## PATENT ABSTRACTS OF JAPAN

(11)Publication number: 2001-142937 (43) Date of publication of application: 25.05.2001

(51) Int.Ci.

G06F 17/50

(21)Application number: 2000-106543

(71)Applicant: NEC CORP

(22) Date of filing:

07.04.2000

(72)Inventor: ASHAR PRANAV

SUBURAJITTO BATACHARIYA

**RAGHUNATHAN ANAND** 

**GUPTA AARTI** 

(30) Priority

Priority number: 1999 414815 Priority date: 08.10.1999 Priority country: US

## (54) SCHEDULING CORRECTNESS CHECKING METHOD AND SCHEDULE VERIFYING METHOD FOR **CIRCUIT**

(57) Abstract:

PROBLEM TO BE SOLVED: To provide a method for checking the correctness of scheduling of a circuit and a method for verifying the schedule of the circuit corresponding to the behavior description of the circuit. SOLUTION: The schedule for the circuit is provided from the behavior description. Concerning the method for checking the correctness of scheduling of the circuit, a loop invariant term is extracted for determining the sufficient set of a non-cyclic thread while a loop is inside the circuit, a symbolic simulation is executed for extracting the loop invariant term, and the equivalency of the non-cyclic thread is proved. Concerning the method for verifying the schedule of the circuit corresponding to the behavior description of the circuit, the schedule thread of possible execution containing the loop is selected from the schedule, a correspondent behavior thread is identified out of the behavior description, the un-conditional equivalency of the schedule thread and the behavior thread is proved, and the operation is repeated concerning all the threads of execution.



## **LEGAL STATUS**

[Date of request for examination]

09.03.2001

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

Copyright (C); 1998,2003 Japan Patent Office

公裁(4) 特許 噩 (Z) (19) 日本国格群庁 (JP)

(11)特許出願公開番号

**特開2001-142937** 

(P2001 - 142937A)

(43)公開日 平成13年5月25日(2001.5.25)

デーフー・(参考) 664G 5B046 11/20 G06F <u>-</u> 做別紀甲 664 G06F 17/50 (51) Int.Cl.<sup>7</sup>

## 審査翻束 未開求 開求項の数43 OL (全37 貝)

-			
(21)出題辞号	特展2000-106543(P2000-106543)	(71) 出版人 000004237	000004237
			日本電気株式会社
(22) ((間日	平成12年4月7日(2000.4.7)		東京都港区芝五丁目7番1号
		(72) 発明者	プラナブ・アシャー
(31)優先権主張条件	(31)優先権主選条号 09/414815		アメリカ合衆国、ニュージャージー
(32)優先日	平成11年10月8日(1999.10.8)		08540 プリンストン、4 インディベン
(33)優先橋主張国	米國 (ns)		デンス ウエイ, エヌ・イー・シー・ユ
			一・エス・エー・インク内
		(74)代理人 100097157	100097157
			<b>介理士 桂木 雄二</b>
			最終買に焼く

回路のスケジューリング正当性チェック方法及びスケジュール検証方法 (54) [発売の名称]

57] [要約]

する方法、及び、回路のどヘイピア配述に対して回路の 【課題】 回路のスケジューリングの正当性をチェック スケジュールを検証する方法を実現する。

チェックする方法は、ループが回路内にあるときに非巡 回スレッドの十分なセットを決定するためにループ不変 項を抽出し、ループ不変項を抽出するためにシンポリッ し、アヘイアア哲説から対応するアヘイアアメレッドを 【解決手段】 回路に対するスケジュールはピヘイピア **クシミュレーションを実行し、非巡回スレッドの等価性** を証明する。回路のどヘイビア記述に対して回路のスケ ジュールを検証する方法は、スケジュールからループを 弘述から得られる。回路のスケジューリングの正当性を 含む可能性のある実行のスケジュールスレッドを選択

類別し、スケジュールスレッド及びどヘイピアスレッド の無条件等価性を証明し、実行のすべてのスレッドにつ

いて以上を繰り返す。

E.41

サークエミグ 空川美原変 イヤール ر ۱۴۱

特許請求の範囲】

記述から得られる場合の当該回路のスケジューリングの 【静泉項1】 回路に対するスケジュールがピヘイビア 正当性をチェックする方法において、

- (a) ループが回路内にあるときに非巡回スレッドの十 分なセットを決定するためにループ不変項を抽出するス
- (b) 前記ループ不変項を抽出するためにシンポリック シミュレーションを実行するステップと、
- (c) 前記非巡回スレッドの等価性を証明するステップ

からなることを特徴とする回路スケジューリング正当性 チェック方法。

【静水項2】 前記どヘイピア配述は、サイクル境界の

導入によって変換されることを特徴とする請求項1配載 【開水項3】 前配ピヘイピア配迹は、演算並べ替えに よって変換されることを特徴とする請求項1配載の方 【群状頃4】 「前覧アヘイアア配送は、ループの原開、

警付け、折畳み及びパイプライン化によって変換される 【請求項5】 前配ピヘイピア配述は、演算の投機実行 ことを特徴とする請求項1記載の方法。

によって変換されることを特徴とする請求項1配載の方 【静永項6】 前記ステップ (c) は、シンボリックシ ミュレーションを用いて実行されることを特徴とする語 【部状版7】 回路のアヘイアア記述に対して回路のス ケジュールを検証する方法において、

水項1配板の方法。

- (8) 値配スケジュールから、ループを含む可能性のあ (4) 位的アヘイアア配派から、対応するアヘイアアス る実行のスケジュールスレッドを選択するステップと、
- (c) スケジュールスレッド及びピヘイピアスレッドの 無条件等価性を証明するステップと、 レッドを餞別するステップと、
- (4) 実行のすべてのスレッドについて前配ステップ (a) ~ (c) を繰り返すステップと、
- 「請求項8】 前配スケジュールは、スケジュール状態 **雪移グラフとして指定されることを特徴とする請求項7** からなることを特徴とする回路スケジュール検証方法。

グラフとして指定されることを特徴とする請求項7 配検 **哲的アヘイピアは、アヘイアア状態避移** [開水項9]

配載の方法。

(i) 前記スケジュールスレッドをスケジュール構造グ ラフに変換するとともに前配アヘイピアスレッドをどへ イビア構造グラフに変換するステップと、 前記ステップ (c) は、

8 (ii) 前記スケジュール構造グラフと前記ピヘイピア

3

特開2001-142937 (P2001-142937A)

構造グラフの等価性をチェックするステップと.

【翻求項11】 回路のピヘイビア記述に対して回路の「 からなることを特徴とする翻求項1記載の方法。 スケジュールを検証する方法において、

- (a) スケジュールをスケジュール状態遷移グラフとし て指定するステップと、
- (b) 同路のピヘイピアをピヘイピア状態選移グラフと して表現するステップと、
- ケジュールスレッドを選択するステップと、

(c) 前配スケジュール状態圏移グラフから、実行のス

- (d) 前記ピヘイピア状態遷移グラフから、対応するピ ヘイピアスレッドを識別するステップと
- (e) 前記スケジュールスレッドをスケジュール構造グ ラフに変換するとともに前配ピヘイピアスレッドをピヘ イピア構造グラフに変換するステップと、
- (1) 前記スケジュール構造グラフと前記ピヘイピア構 **治グラフの等価性をチェックするステップと、**
- (g) 実行のすべてのスレッドについて前記ステップ (c) ~ (l) を繰り返すステップと、
- からなることを特徴とする同路スケジュール検証方法。 【請求項12】 前記ステップ(I)は、 8
- (i) 前記ピヘイピア状態遷移グラフ内の各ノードが核 ノードの揺移ファンイン内のナベトのノードの後にのみ 現れるように、前記ピヘイピア構造グラフ内のすべての ノードを含む順序セット8FFIを作成するステップ
- 一ドの植移ファンイン内のすべてのノードの後にのみ現 れるように、前記スケジュール構造グラフ内のすべての (ii) 前記ピヘイピア構造グラフ内の各ノードが核ノ ノードを含む順序セット8552を作成するステップ
- (1111) arr1をたどり、ピヘイビア構造グラン内 の基底変数を識別するステップと
  - (iv) ピヘイピア構造グラフ内の非基底変数を基底変
- (v) スケジュール構造グラフ内の入力ノードに対する 敬で表すステップと、
- (vi) arr2をたどり、arr2内の各ノードを処 **型して、スケジュール構造グラフの人力からスケジュー** 等価性リストを構成するステップと
- ル構造グラフの川力へ等価性リストを伝像させるステッ (vii) uをピヘイピア構造グラフ内の信号の観別で
- とし、cを等価性の条件を表す二分次定ダイヤグラムで **ードで等価性が確定したかどうか、及び、対応する条件** c がa r r 2内のプライマリ出力ノードに対するトート c) であり、ピヘイピア構造グラフ内の対応する川カノ あるとして、各等価性リスト内のエントリは対(u.
- (viii) arr2内のすべての出力ノードについて ロジーであるかどうかをチェックするステップと、 前記ステップ(vii)を繰り返すステップと、

(ix) すべての出力ノードが等値であることがわかっ た場合に等価性を見つけるステップと、

【請求項13】 実行の巡回スレッドを有する可能性の ある回路のスケジュールと核回路のピヘイピアとの団の からなることを特徴とする間求項11記載の方法。 等価性を検証する方法において、

- (a) スケジュールをスケジュール状態選移グラフとし て表現するステップと、
- (16) ピヘイピアをピヘイピア状態選移グラフとして数 見するステップと、
- (c) 前記スケジュール状態選移グラフ内の強逆結成分
- (d) 各強逆結成分内の終了ノードを識別するステップ を磔別するステップと、
- (e) 崩割スケジュール状態顕移グラフをつぶして、歯 紀始連結成分を通らないサブパスを併合するステップ
- (1) 以前に選択されていないパスを選択するステップ
- (g) 前記ステップ (f) で強択されたパスに対する構
  - (h) 選択されたパスを列挙するのに必要なすべての状 **顔遷移決定をカプセル化するパスシグナルを生成するた** 造RTL回路を取得するステップと、
    - (i) パスシグナルを用いて、脚約されたシンボリック シミュレーションを実行してピヘイピア状態通移グラフ 内の対応するパスを織別し、抜パスに対する構造RTL めの回路を構造RTL回路に追加するステップと、
- (j) 顕状されたパスにおいて、以前に選択されていな い強連結成分を選択するステップと、
- る不変項を、対応セットのリストとして抽川するステッ (k) 遊択されたパス内の選択された強連結成分に対す
- (1) 対応セットのリストから1つの対応セットを選択 するステップと、
- ュアーションの強逆結成分カットにおいて得られる変数 (m) 選択された対応セットが、前のシンボリックシミ 姓のより小さい場合に、ツンボリックシミュレーション を再実行するステップと、
  - (n) 対応セットのリスト内の各対応セットについて前 記ステップ (i) ~ (m) を繰り返すステップと、
    - (o) 川力等価性条件が、パス条件以外の条件付きであ (p) 崩魁ステップ (o) で崩配出力等価性が条件付き るかどうかをテストするステップと、
- (4) 選択されたパス内のす。ペイの強連結成分について である場合に非導価性を報告してこの方法を終了するス
- (1) 終了点が高々3度現れるようにルートからシンク へのすべてのパスについて前記ステップ (f) ~ (q) 前記ステップ(j)~(p)を繰り返すステップと、

からなることを特徴とする前配回路のスケジュールとビ ヘイピアとの間の等価性を検証する方法。 を繰り返すステップと、

【請求項14】 前記ステップ (i) の勘約されないシ ンボリックシミュレーションは、 (i) ピヘイピア状態遷移グラフの始状態を許容パスリ ストに割り当てるステップと、 (ii) 許容パスリスト内で以前に訪れていない状態を (iii) ピヘイピア構造RTLを生成するステップ 選択するステップと、

(iv) 非解釈シンボリックシミュレーションを実行し て、スケジュール構造RTL及びピヘイピア構造RTL 内の対応する佰号を識別するステップと、 (v) 選移条件とパスシグナルの論理相がゼロでない場 合に、状態Sjの新しいコピーを許容パスに追加するス デップと、

(vi) SiからSjへの各出避移ごとに前配ステップ

(v) を繰り返すステップと、

状態のインスタンスとなるまで、すべての訪れていない (v i i) 許容パス内に残る訪れていない状態のみが終 状値について前記ステップ (i i i i) ~ (v i) を繰り 返すステップと 8

からなるプロセスを用いて実行されることを特徴とする 清水項13配級の方法。

**【静米項15】 前記ステップ (k) において、不変項** は、各ループごとに、 (i) 各カットが削配ループの各実行の境界における変 数値を安すような、スケジュール内のパスの構造RTL 回路内の3個のカットを織別するステップと、

(ii) ピヘイピアにおけるパスの構造RTL回路内の 対応するカットを臨別して、第1と第2のカットの間の サブ回路と、第2と第3のカットの間のサブ回路が同型

(iii) スケジュール及びどヘイピアのRTL回路に おける対応するカットの各対における変数どうしの間の であることをチェックするステップと、

**労価関係を職別するステップと、** 

(iv) 最後のカットと最後の前のカットとの間の等価 つ、最後のカットにおける等価関係が、最後の前のカッ トにおける等価関係のサブセットである場合、最後の前 実行について2つのRT-L回路を展開して、前記ステッ のカットにおける华恒関係を依頼し、1つ以上のループ 関係が同一であるかどうかをチェックするステップと、 (v) 前跑ステップ (iv) の関係が同一でなく、か プ(iii)から繰り返すステップと、

追加し、1つ以上のループ実行について2つのRTL回 (vi) 前記ステップ (iv) の関係が同一でなく、か トにおける等価関係のサブセットでない場合、最後の前 のカットにおける等価関係を、等価関係セットの集合に つ、最後のカットにおける等価関係が、最後の前のカッ

路を履開して、前配ステップ(iii) から繰り返すス

(v i i) 煎配ステップ(i v)の関係が同一である場 合、最後のカットにおける等価関係を、等価関係セット の集合に追加するステップと、

(viii) 特価関係セットの集合内で、他のエントリ のスーパーセットであるすべてのエントリを削除するス テップと、 (1x) 毎価関係セットの最終集合を、不変項の所留の からなるプロセスを用いてループから抽出されることを 特徴とする請求項13記載の方法。 **集合として指定するステップと**、

【替米項16】 回路に対するメケジュートがかくイア ア配述から得られ、回路のスケジューリングの正当性を チェックするシステムにおいて、

ループが存在するときに非巡回スレッドの十分なセット を決定するループ不変項加出器と、 非巡回スレッドの等価性を証明する等価性証明器と、

前記ループ不変項を抽出するシンボリックシミュレータ

からなることを特徴とする回路のスケジューリングの正 【請求項17】 前記ピヘイピア記述は、サイクル境界 当性をチェックするシステム。

の導入によって変換されることを特徴とする請求項16 配収のシステム。

【静水項18】 前記ピヘイピア記述は、放算並べ替え によって変換されることを特徴とする翻水項16配載の 【韶永項19】 前配ピヘイピア記述は、ループの展 システム。

期、 巻付け、 折畳み及びパイプライン化によって変換さ 【開水項20】 前配アヘイアア配述は、演算の投機実 れることを特徴とする請求項16配線のシステム。

**行によって変換されることを特徴とする請求項16配載** のシステム。

【謝水項21】 回路のピヘイピア記述に対して回路の スケジュールをスケジュール状態避移グラフとして指定 回路のピヘイピアをピヘイピア状態遜移グラフとして指 前記スケジュール状態選移グラフから、実行のスケジュ **ールスレッドを選択するスケジュールスレッドセレクタ** 定するピヘイピア状態器移グラフジェネレータと、 するスケジュール状態選移グラフジェネレータと、 スケジュールを検証するシステムにおいて、

前記ピヘイピア状態選移グラフから、対応するピヘイビ 前記スケジュールスレッドをスケジュール構造グラフに 変換するとともに前配ピヘイピアスレッドをピヘイピア アスレッドを避択するピヘイピアスレッドセレクタと、 構造グラフに変換するコンパータと、

前配スケジュール構造グラフと前配ピヘイピア構造グラ フの等価性をチェックする等価性チェッカと、

からなることを特徴とする回路のピヘイピア記述に対し て回路のスケジュールを検証するシステム。 プロセッサ及びメモリを有するコンピュータシステムに エックするための

回路に対するスケジュールは、ピヘイピア配述から得ら

ループ不変項を抽出するためのシンボリックシミュレー 前記メモリは、前記コンピュータシステムが前記チェッ ループが存在するときに非巡回スレッドの十分なセット クを実行することを可能にする命令を含み、眩命令は を決定するためにループ不変項を抽出する命令と、 ションの色化か、

を含むことを特徴とする回路のスケジューリングの正当 性をチェックするためのコンピュータシステム。 非巡回スレッドの等価性を証明する命令と、

【翻求項23】 前記ピヘイピア記述は、サイクル境界 の導入によって変換されることを特徴とする静水項22 記載のコンピュータシステム。

によって変換されることを特徴とする間水項22記載の 【樹水項24】 前配ピヘイピア配述は、複算並べ皆え コンピュータシステム。

期、卷付け、折畳み及びパイプライン化によって変換さ れることを特徴とする制水項22配載のコンピュータシ 【翻水項25】 「前配ドヘイドア配当は、ループの展 ステム。

【請求項26】 前配ピヘイピア記述は、演算の投機実 **行によって変換されることを特徴とする耐水項2<sup>2</sup>2配敵** のコンピュータシステム。

【請求項27】 回路のどヘイビア鮎近に対して回路の スケジュールを検証するための、プロセッサ及びメモリ を有するコンピュータシステムにおいて、 前記メモリは、前記コンピュータシステムが前記検証を スケジュールをスケジュール状態避移グラフとして指定 **実行することを可能にする命令を含み、核命令は、** 

回路のピヘイピアをピヘイピア状態強移グラフとして安 現する命令と、 する命令と、

前記スケジュール状億選移グラフから、実行のスケジュ **ールスレッドを選択する命令と、** 

前配ピヘイピア状態道移グラフから、対応するピヘイビ

前記スケジュールスレッドをスケジュール構造グラフに アスレッドを選択する命令と、

前配スケジュール構造グラフと前配ピヘイピア構造グラ 変換するとともに 前記 どヘイピアスレッドをどヘイビア 構造グラフに変換する命令と、

実行のすべてのスレッドについて繰り返す命令と、 フの等価性をチェックする命令と

を含むことを特徴とする、回路のピヘイピア記述に対し

3

特 開 2001-142937 (P2001-142937A)

て回路のスケジュールを検証するためのコンピュータシ

【結水瓜28】 回路のアヘイアア配添に対して回路の スケジュールを慎証するための、プロセッサ及びメモリ 前記メモリは、前記コンピュータシステムが、 を有するコンピュータシステムにおいて、

- (n) スケジュールをスケジュール状態遷移グラフとし て指定するステップと、
- (c) 前記スケジュール状態遷移グラフから、実行のス (4) 国路のアヘイアアをアヘイアア状態過移グランと して安児するステップと、
- (4) 煎部ピヘイピア状態強移グラフから、対応するビ ケジュールスレッドを選択するステップと、
- (c) 航記スケジュールスレッドをスケジュール構造グ ラフに変換するとともに前配ピヘイピアスレッドをピヘ ヘイピアスレッドを職別するステップと、 イビア構造グラフに変換するステップと、
- (「) 前記スケジュール構造グラフと前配ピヘイピア構 造グラフの等価性をチェックするステップと、
  - (g) 実行のすべてのスレッドについて前配ステップ
    - (c) ~ (l) を繰り返すステップと
- 【請求項29】 前配命令は、前配コンピュータシステ を実行することを可能にする命令を含むことを特徴とす る、回路のどへイビア記述に対して回路のスケジュール を検証するためのコンピュータンステム。

(i) 前配ピヘイピア状態選移グラフ内の各ノードが該

- **現れるように、前配ピヘイピア構造グラフ内のすべての** ノードの株移ファンイン内のすべてのノードの後にのみ ノードを含む順作セットarr1を作成するステップ
  - ードの推移ファンイン内のすべてのノードの後にのみ現 れるように、前配スケジュール構造グラフ内のすべての (ii) 前記ピヘイピア構造グラフ内の各ノードが抜ノ ノードを含む粕/作セットョェァ2を作成するステップ
- (iii) \*\* ロ に 1をたどり、ピヘイピア構造グラフ内 の基底変数を離別するステップと、
- (iv) ピヘイピア構造グラフ内の非基底変数を基底変 数で要すステップと、
- (v) スケジュール構造グラフ内の入力ノードに対する 等価性リストを構成するステップと、
- (vi) arr2をたどり、arr2内の各ノードを処 ル構造グラフの川力へ等価性リストを伝搬させるステッ 型して、スケジュール構造グラフの入力からスケジュー
- あるとして、ピヘイピア構造グラフ内の対応する出力/ na 配ステップ (i)~ (m)を繰り返すステップと、 あり、cは特価性の条件を表す二分決定ダイヤグラムで であり、uはピヘイピア構造グラフ内の信号の臨別子で (vii) 各等価格リスト内のエントリは対 (u, c)

- **ードで等価性が確定したかどうか、及び、対応する条件** cがarr2内のプライマリ出力ノードに対するトート (viii) arr2内のすべての出力ノードについて ロジーであるかどうかをチェックするステップと、
- (ix) すべての出力ノードが挙仰であることがわかっ 前記ステップ(vii)を繰り返すステップと、
- を用いて前記ステップ(1)を実行することを可能にす る命令をさらに含むことを特徴とする請求項28に配做 た場合に等価性を見つけたとするステップと、 のコンピュータシステム。
  - 【請求項30】 回路のスケジュールと該回路のピヘイ ピアとの間の等価性を検証するための、プロセッサ及び
    - 前記スケジュール及び前記ピヘイピアは、実行の巡回ス メモリを有するコンピュータシステムにおいて、 レッドを有する可能性があり、
- 前記メモリは、前配コンピュータシステムが、
- (a) スケジュールをスケジュール状態選移グラフとし て表現するステップと、
- (b) ピヘイピアをピヘイピア状個過移グラフとして表 現するステップと、
- (c) 前記スケジュール状態遷移グラフ内の強遊結成分
  - (d) 各強連結成分内の終了ノードを識別するステップ を臨別するステップと、
- (e) 前記スケジュール状態遷移グラフをつぶして、前 配強連結成分を通らないサブパスを併合するステップ
  - (1) 以前に選択されていないパスを選択するステップ
- (g) 前記ステップ (1) で選択されたパスに対する情 造RTL回路を取得するステップと、
  - (h) 選択されたパスを列挙するのに必要なすべての状 **閻邏移決定をカプセル化するパスシグナルを生成するた** めの回路を構造RTL回路に追加するステップと、
    - (1) パスシグナルを用いて、制約されたシンボリック シミュレーションを実行してピヘイピア状態選移グラフ 内の対応するパスを購別するステップと、
- (j) 選択されたパスにおいて、以前に選択されていな い強連結成分を選択するステップと、
- (k) 選択されたパス内の選択された強連結成分に対す る不変項を、対応セットのリストとして抽出するステッ
- (1) 対応セットのリストから1つの対応セットを選択 するステップと、
- ュレーションの強連結成分カットにおいて得られる変数 対応より小さい場合に、シンボリックシミュレーション (m) 遊択された対応セットが、前のシンボリックシミ を再実行するステップと、
- (n) 対応セットのリスト内の各対応セットについて前

(0) 出力等価性条件が、非等価性を報告するパス条件 以外の条件付きであるかどうかをテストするステップ

(p) 前記ステップ (o) で前配出力等価性が条件付き である場合にこの検証を終了するステップと、

(d) 選択されたパス内のすべての強速結成分について **前配ステップ(j) ~(p) を繰り返すステップと、** 

を繰り返すステップとを用いて前配検証を実行すること 核回路のピヘイピアとの間の等価性を検託するためのコ を可能にすることを特徴とする、回路のスケジュールと (r) 終了点が高々 3 度現れるようにルートからシンク へのすべてのパスについて前配ステップ (f)~(q) ンピュータシステム。

【請求項31】 前配命令は、前配コンピュータシステ

- (1) ビヘイピア状態遷移グラフの始状態を許容パスリ ストに割り当てるステップと
- (ii) 許容パスリスト内で以前に訪れていない状態を 徴択するステップと、
- (iii) ピヘイピア構造RTLを生成するステップ
- (iv) 非解釈シンボリックシミュレーションを実行し て、スケジュール構造RTL及びピヘイピア構造RTL
- (v) 通移条件とパスシグナルの論型積がゼロでない場 合に、状態Sjの新しいコピーを許容パスに迫加するス 内の対応する信号を臨別するステップと、
- (vi) SjからSjへの各出選移ごとに前配ステップ (v) を繰り返すステップと、
- (vii) 許容パス内に残る助れていない状態のみが終 状態のインスタンスとなるまで、すべての訪れていない を可能にする命令をさらに含むことを特徴とする請求項 状態について前記ステップ (iiii) ~ (vi) を繰り 返すステップとを用いてステップ (i) を実行すること 30に記載のコンピュータシステム。

【請求項32】 前配命令は、前配コンピュータシステ ムが、各ループごとに、

- (1) 各カットが前配ループの各実行の境界における変 殺値を表すような、スケジュール内のパスの構造RTL 回路内の3個のカットを贈別するステップと、
- (ii) ピヘイピアにおけるパスの構造RTL回路内の 対応するカットを離別して、第1と第2のカットの間の サブ回路と、第2と第3のカットの間のサブ回路が同盟 であることをチェックするステップと、
- (iii) スケジュール及びどヘイピアのRTL回路に おける対応するカットの各対における変数どうしの間の 等価関係を識別するステップと、
- (iv) 最後のカットと最後の前のカットとの間の等価 (v) 前記ステップ (i v) の関係が同一でなく、か 関係が同一であるかどうかをチェックするステップと、

9

特 (別2001-142937 (P2001-142937A)

つ、最後のカットにおける等価関係が、最後の前のカッ トにおける等価関係のサブセットである場合、最後の前 実行について2つのRTL両路を展開して、前起ステッ のカットにおける等価関係を依乗し、1 つ以上のループ ブ(iii)から繰り返すステップと、 (vi) 前記ステップ(iv)の関係が同一でなく、か・ トにおける等価関係のサブセットでない場合、最後の前 のカットにおける等価関係を、等価関係セットの集合に 道加し、1つ以上のルーブ実行について2つのRT上回 つ、最後のカットにおける等価関係が、最後の前のカッ 略を展開して、前配ステップ(iii)から繰り返すス

(vii) 前記ステップ (iv) の関係が同一である場 合、最後のカットにおける等価関係を、等価関係セット の集合に追加するステップと、

テップと

(viii) 等価関係セットの集合内で、他のエントリ のスーパーセットであるすべてのエントリを削除するス テップと、 (1x) 等価関係セットの最終集合を、不変項の所母の を用いて前記ステップ(k)を実行することを可能にす 集合として指定するステップと、 8

る命令をさらに含むことを特徴とする請求項30記載の コンピュータシステム。

【語水項33】 コンピュータが回路のスケジューリン グの正当性をチェックすることを可能にするコンピュー タコードを含むコンピュータ可能媒体を有するコンピュ ータプログラム製品において、

前配コンピュータコードは、

回路に対するスケジュールは、アヘイピア記述から得ら

ループが存在するときに非巡回スレッドの十分なセット を決定するためにループ不変項を抽出するコンピュータ 7411

ループ不変項を抽出するためのシンボリックシミュレー ションのコンピュータコードと、

非巡回スレッドの等価性を原列するコンピュータコード とを含むことを特徴とする、コンピュータが回路のスケ ジューリングの正当性をチェックすることを可能にする コンピュータコードを含むコンピュータ可能媒体を有す

【請求項34】 | 前配ピヘイピア配派は、サイクル境界 の導入によって変換されることを特徴とする請求項33 るコンピュータプログラム製品。

によって変換されることを特徴とする語求項33記載の 【請求項35】 前記ピヘイピア記述は、演算並べ替え 的戯のコンピュータプログラム製品。 コンピュータブログラム製品。

れることを特徴とする請求項33配板のコンピュータブ 明、簪付け、折畳み及びパイプライン化によって変換さ 【請求項36】 前配ピヘイピア記述は、ループの限

ログラム戦品。

げによって変換されることを特徴とする群求項33記載 [請求項37] 前配ピヘイピア勧送は、演算の投機実 のコンピュータブログラム製品。

【荫水項38】 コンピュータが回路のビヘイビア記述 るコンピュータコードを含むコンピュータ可能媒体を有 に対して回路のスケジュールを検証することを可能にす するコンピュータプログラム製品において、散コンピュ

**前記コンピュータが、スケジュールをスケジュール状態** 遊移グラフとして指定することを可能にするスケジュー ル状態強移グラフジェネレータコードと、

**近割コンピュータが、回路のピヘイピアをピヘイピア状** 低強移グラフとして指定することを可能にするピヘイビ ア状修選移グラフジェネレータコードと、

前記コンピュータが、前記スケジュール状態遷移グラフ から、実行のスケジュールスレッドを選択することを可 前記コンピュータが、前記ピヘイピア状態選移グラフか 能にするスケジュールスレッドセレクタコードと、

1

にするどへイビアスレッドセレクタコードと、前配コン 構造グラフに変換するとともに前配ピヘイピアスレッド ら、対応するピヘイピアスレッドを選択することを可能 ピュータが、前配スケジュールスレッドをスケジュール をピヘイピア構造グラフに変換することを可能にするコ ンベータコードと、

前記コンピュータが、前配スケジュール構造グラフと前 記ピヘイピア構造グラフの等価性をチェックすることを 可能にする等価性チェッカコードと、

を可能にするコンピュータコードを含むコンピュータ可 からなることを特徴とする、コンピュータが回路のピヘ イビア記述に対して回路のスケジュールを検証すること 散媒体を有するコンピュータプログラム製品。

【排氷項39】 コンピュータが回路のアヘイピア記述 に対して回路のスケジュールを検証することを可能にす るコンピュータコードを含むコンピュータ可能媒体を有 演乱コンピュータコードは、追記コンピュータが、 するコンピュータプログラム製品において、

- (\*) スケジュールをスケジュール状態選移グラフとし
- (b) 回路のピペイピアをピペイピア状態選移グラフと て指定するステップと、

して表現するステップと、

- (c) 崩乱スケジュール状態遷移グラフから、実行のス (4) 前記ピヘイピア状態強移グラフから、対応するピ ケジュールスレッドを選択するステップと ヘイピアスレッドを識別するステップと、
- ラッに変換するとともに前記ピヘイピアスレッドをピヘ (e) 前記スケジュールスレッドをスケジュール構造グ イビア構造グラフに変換するステップと、
- (1) 前記スケジュール構造グラフと前記ピヘイピア構 造グラフの等価性をチェックするステップと、

(g) 実行のすべてのスレッドについて前記ステップ

(c) ~ (f) を繰り返すステップと、

ールを検証することを可能にするコンピュータコードを を実行することを可能にすることを特徴とする、コンピ ュータが回路のどへイビア記述に対して回路のスケジュ 含むコンピュータ 可酸媒体を有するコンピュータプログ ラム製品。

【請水項40】 前記コンピュータコードは、前記コン ピュータが

- (i) 前紀ピヘイピア状態選移グラフ内の各ノードが抜 現れるように、前記ピヘイピア構造グラフ内のすべての ノードの推移ファンイン内のすべてのノードの後にのみ ノードを含む順序セットarr1を作成するステップ
- れるように、前記スケジュール構造グラフ内のすべての (ii) 前記ピヘイピア構造グラフ内の各ノードが抜ノ **一ドの推移ファンイン内のすべてのノードの後にのみ現** ノードを含む順序セットBrr2を作成するステップ
  - (iii) urr1をたどり、ピヘイピア構造グラフ内
- (iv) ピヘイピア構造グラフ内の非基底変数を基底変 の基底変数を職別するステップと、
  - 散で表すステップと、
- (v) スケジュール構造グラフ内の入力ノードに対する 箏価性リストを構成するステップと、
- (vi) arr2をたどり、arr2内の各ノードを処 **卑して、スケジュール構造グラフの入力からスケジュー ル構造グラフの出力へ等価性リストを伝搬させるステッ**
- であり、uはどヘイビア構造グラフ内の信号の織別子で あり、cは等価性の条件を数す二分決定タイヤグラムで **ードで等価性が確定したかどうか、及び、対応する条件** (vii) 各等価性リスト内のエントリは対 (u, c) あるとして、どヘイビア構造グラフ内の対応する出力ノ c がa r r 2内のプライマリ出力ノードに対するトート ロジーであるかどうかをチェックするステップと、
- (viii) arr2内のすべての出力ノードについて 前記ステップ(vii)を繰り返すステップと、
- (ix) すべての出力ノードが等価であることがわかっ た場合に等価性を見つけたとするステップと、
  - を用いて前記ステップ (1) を実行することを可能にす ることを特徴とする請求項39配載のコンピュータプロ グラム製品

豚回路のビヘイピアとの間の等価性を検証することを可 【船水項41】 コンピュータが回路のスケジュールと 能にするコンピュータコードを含むコンピュータ可能媒 前記スケジュール及び前配ピヘイピアは、実行の巡回ス 体を有するコンピュータプログラム製品において、

(a) スケジュールをスケジュール状態避移グラフとし 前記コンピュータコードは、前記コンピュータが、 レッドを有する可能性があり、

13 に表現するステップと

- (b) どヘイピアをピヘイピア状態遷移グラフとして扱 見するステップと
- (c) 前記スケジュール状態遷移グラフ内の強連結成分 2 職別するステップと
- (d) 各強連結成分内の終了ノードを識別するステップ
  - (e) 前記スケジュール状態避移グラフをつぶして、前 **印油連結成分を通らないサブパスを併合するステップ**

(f) 以前に選択されていないパスを選択するステップ

- (g) 前記ステップ (I) で選択されたパスに対する構
- (h) 選択されたパスを列挙するのに必要なすべての状 **街遜移決定をカプセル化するパスシグナルを生成するた** 造RTL回路を取得するステップと、
- **内の対応するパスを識別し、抜パスに対する構造RTL** (i) パスシグナルを用いて、制約されたシンボリック シミュレーションを実行してピヘイピア状態遷移グラフ りの回路を構造RTL回路に追加するステップと、 回路を取得するステップと、
- (j) 選択されたパスにおいて、以前に選択されていな い強連結成分を選択するステップと、
- (k) 選択されたパス内の選択された強連結成分に対す る不変項を、対応セットのリストとして杣出するステッ
- (1) 対応セットのリストから1つの対応セットを選択 するステップと、
- ュレーションの強連結成分カットにおいて得られる変数 4店より小さい場合に、シンボリックシミュレーション (田) 選択された対応セットが、前のシンボリックシミ を再実行するステップと、
- (n) 対応セットのリスト内の各対応セットについて前 記ステップ (i) ~ (m) を繰り返すステップと、
  - (0) 出力等価性条件が、パス条件以外の条件付きであ るかどうかをテストするステップと、
- (p) 前配ステップ (o) で前配出力等価性が条件付き である場合に非等価性を報告してこの方法を終了するス デップと、
- (d) 選択されたパス内のすべての強連結成分について (ェ) 棒丁点が高々 3度現れるようにルートからシンク 前記ステップ (j) ~ (p) を繰り返すステップと、
  - を実行することを可能にすることを特徴とする、コンピ ュータが回路のスケジュールと該回路のどへイビアとの 間の等価性を検証することを可能にするコンピュータコ **一ドを含むコンピュータ可説媒体を有するコンピュータ** へのすべてのパスについて前記ステップ (f) ~ (d) を繰り返すステップと、
- 【請求項42】 前記コンピュータコードは、前記コン so プログラム製品。

テップと、

8

14

ピュータが、

- (i) ピヘイピア状態選移グラフの始状態を許容パスリ ストに割り当てるステップと、
- (ii) 許容パスリスト内で以前に訪れていない状態を 選択するステップと、
- (iii) ピヘイピア構造RTLを生成するステップ
- て、スケジュール構造RTL及びピヘイピア構造RTL (iv) 非解釈シンボリックシミュレーションを実行し 内の対応する信号を識別するステップと、
- (v) 選移条件とパスシグナルの輪型積がゼロでない場 合に、状態Sjの新しいコピーを許容パスに追加するス テップと,
- (vi) SiからSjへの各出遺移ごとに前記ステップ (v) を繰り返すステップと、
- 状態のインスタンスとなるまで、すべての訪れていない (vii) 許容パス内に残る別れていない状態のみが終 状態について前記ステップ (iiii) ~ (vi) を繰り 返すステップと、
- を用いて、前配ステップ(i)の制約されないシンポリ ックシミュレーションを実行することを可能にすること を特徴とする謝水項41配載のコンピュータブログラム
- 【脳水項43】 前記コンピュータコードは、前記コン ピュータが、各ループごとに、
- (i) 各カットが前記ループの各実行の境界における変 数値を表すような、スケジュール内のパスの構造RTL 回路内の3個のカットを敵別するステップと、
- (ii) アヘイアアにおけるパスの結治Rに1回程左の 対応するカットを韓別して、第1と第2のカットの街の サブ回路と、第2と第3のカットの間のサブ回路が河型 であることをチェックするステップと、
- (iiii) スケジュール及びアヘイピアのRTL回路に おける対応するカットの各対における変数とうしの間の 等価関係を職別するステップと、
- (i v) 最後のカットと最後の前のカットとの間の等価 関係が同一であるかどうかをチェックするステップと、
- つ、最後のカットにおける等価関係が、最後の前のカッ トにおける等価関係のサブセットである場合、最後の前 のカットにおける等価関係を破棄し、1 つ以上のルーブ **実行について2つのRTL回路を展開して、前記ステッ** (v) 植記ステップ (iv) の関係が同一でなく、か
- トにおける等価関係のサブセットでない場合、最後の前 (vi) 前記ステップ (iv) の関係が同一でなく、か つ、最後のカットにおける等価関係が、最後の前のカッ のカットにおける等価関係を、等価関係セットの集合に **適加し、10以上のループ実行について20のRTL**回 路を服開して、前配ステップ(iii)から繰り返すス プ(iii)から繰り返すステップと、

のスーパーセットであるすべてのエントリを削除するス (viii) 等値図係セットの集合内で、他のエントリ の集合に追加するステップと、

デップと、

(ix) 等価関係セットの最終集合を、不変項の所望の を用いて前記ステップ (k) で不変項を加出することを 可能にすることを特徴とする静水項41 配載のコンピュ 集合として指定するステップと、

[発明の詳細な説明]

ータプログラム製品。

[000]

中に実行されることがある。スケジューリングをチェッ

クしようとする検証ツールにとって最小限の要件は、こ れらの変換をそのスコープ(有効範囲)に含むことであ

合成におけるスケジューリングステップの検証(verific る。特に、本発明は、ループと、スケジューリング中に [1. 1 発明の属する技術分野] 本発明はハイレベル ングとともに実行される可能性が高いすべての代表的な 変換を含むスケジューリング検証のための新規技術にあ **支行されるさまざまなループ変換を扱うことが可能な検** ation)に関する。本発明の主要な焦点は、スケジューリ [発明の属する技術分野] [1. 発明の詳細な説明] 証技術を提供する。

合せ輪理の検証は、初期ネットリスト仕様に対して最終 **協理ネットリストの妥当性の検証を行う必要があるため** を検証するためのツールも必要となる。本発明は、検証 れているように、シミュレーションは、正当性(correct より、高速合成が可能になるのに加えて、再使用の観点 びその他の多くの会社から提供されるツールを用いた組 レジスタトランスファレベル (RTL) のネットリスト を実行する技術を改飾するためのものである。よく知ら **位証ストラテジとして十分ではあり得ない。そこで、フ** [従来の技術] [1.2 従来の技術] 回路を出荷する までの時間を短縮する手段として、ハイレベル仕様から の合成が重要であることはよく認識されている。これに **に、く人フバラ・アヘ人 アレ (慰存) 既消かの命のわめ** からもより有利となる。Chrysalis(R)、Synopsys(R)及 noss)を保証しないにも拘わらず時間がかかるために、 [0002]

の知識を活用しなければならない。 実際、スケジューリ [0003] 初期ピヘイピア仕様から最終RTLを実現 えられている場合、入力として単に2つの大幅に異なる **奉いに、合成自体は、自動ツールを用いてなされるか手** ング、リソース割当て及びレジスタ代入のような明確に 区分された基本的なステップからなる共通の基本フロー するために適用される変換のスコープ (有効範囲) が与 的でなされるかにかかわらず、一般的に、スケジューリ に従う。検証方法が実際的であるためには、このフロー は、すべての実際的な目的で実現可能なわけではない。 レベルでの記述をとるブラックボックス検証システム ォーマル検証の方法論が必要となる。

すために、放算並べ替え、ループ展開、投機実行 (spec いストラテジである。最終散計の品質が多少犠牲になっ n せることによって行われる。さまざまな散計要件を満た ングやレジスタ代入のようなステップどうしの間の区分 をそのまま保持することは、「検証のための設計」の良 の検証は、合成プロセス全体を検証するよりは容易であ **糞である。同期設計では、これは、資算に状態を対応さ** 【0004】ハイレベル合成フロー中の個々のステップ ulative execution) 等のような変換が、このステップ たとしても、合成プロセスははるかに検証容易になる。 は、タイムスタンプを資算(operation)に割り当てる作 るものの、決して簡単ではない。 スケジューリングと

[0005] 本男哲敬において、シンボリックシミュレ d)」という用語は、この場合、標準の算格消算のような はなく、入力リスト及び資算名が転送されることを意味 **一ションとは、回路を通して、変数値ではなく、変数を** 複雑な演算に遭遇したときに、入力のブール演算の値で 伝版させる手続きを含意する。「非解釈(uninterprote

従来、ハイレベル記述から生成される設計を検証するた [0006] 1.2.1 関連する研究

hardware verification", in Proc. Design Automation Conf., pp. 375-381, June 1979、を参照。しかし、Darr めのいくつかの技術が提案されている。プログラム及び になされた。代表的なものとして、J. Darringer, "The リングを検証するという場合に応用を限定している。派 ハードウェアの検証のためのシンボリックシミュレーシ ョンに関するかなりの研究活動が70年代及び80年代 application of program verificationtechniques to ingerの研究及びそこから原生した研究は、スケジュー 生した研究の一部は、

ification with ADLIBand SUL", in Proc. Design Auto . W. Cory, "Symbolic simulation for functional ver mation Conf., pp. 82-89, June 1981 · V. Pitchumani and E. Stabler, "A formal method f or computer design verification", in Proc. Design Automation Conf., pp. 809-814, June 1982

に見られる。

な制限は、シンポリックシミュレータがチェックを実行 らない対応点 (Darringerの用語では制御点(control po 2 つのハードウェア配述を比較する際に、不変項は、一 **方の記述の完全な状態が他方の状態と一致しなければな** so int)) である。スケジューリングの場合、シミュレータ [0007] 重要な点として、Darringerの研究の主要 するための不変項(invariant)を散けることをユーザに 思求していたことである。実際、知られているように、

特別2001-142937 (P2001-142937A)

9

abolic program for checking functional and timingc International Symposium on High Level Synthesis, p にこの情報を提供するために、ユーザは、例えば、合成 ツールによって実行されるループ変換の詳細な知識を有 に、このような要求は、検証の目的に部分的に反するこ とになる。また、ユーザが対応点を提供する場合、完全 性の問題は未解決のままとなる。制御点どうしの聞の中 間信号間の対応を検出し、それを利用して、制御点にお M. C.-T. Chen and A. Parker, "A hybrid numeric/sy ける同型(isomorphism)のためにチェックすべき式を単 ompatibility of synthesized design", in Proc. The する必要がある。このような要求は困難である。さら 純化する追加能力を有する同じ基本的なアルゴリズム

とである。R. A. Bergamaschi and S. Raje, "Observab チを提案している。S. Winato, "Generation of BDDs f ットを提案した。J. Gong, C. T. Chen, and K. Kucukc 時点で観測されなければならないときにどのようにすれ ults", IEEE Design & Test ofComputers, vol.8, pp.4 ざまなステップをチェックするための規則スイートのセ akar, "Multi-dimensional rule checking for high-le le timewindows: Verifying high-level synthesis res 【0008】他のいくつかの関連する文献についてもこ こで説明する。Minatolt、2つのハードウェア記述どう しの関の等価性を確かめるためのBDD (Binary Decis ionDiagram:二分決定ダイヤグラム)に基づくアプロー rom hardware algorithm descriptions", in Proc. Int. Conf. Computer-Aided Design, pp. 644-649, Nov. 199 る。さらに、ルーブは、すべての変数に対するBDDが 追加原間で変化しなくなるまで各ループを展開すること によって処理される。この方法は、算術関数を表現する 際のBDDの制限と、ループ終了条件が満たされるまで ループを明示的に限開する必要があることとによる欠点 vel design verification", in Proc. Int. High-level **照。しかし、彼らの箏価性チェッカは、構造同型をチェ** の貢献は、2 つの配述における対応する信号が相異なる ば等価性チェックを実行することができるかを示したこ がある。Gong et al. は、ハイレベル合成におけるさま Design Validation & Test Wkshp., Nov. 1997、を畚 ックすることに制限されていた。Bergamaschiand Raje は、追加変数の使用により直線的なコードに変換され 6、を参照。このアプローチでは、すべての条件分岐 0-50, Apr. 1997、を参照。

[0009]最近では、検証において算術及び関御算術 相互作用をモデル化するためのいくつかの技術が提案さ · K. T. Cheng and A. S. Krishnakumar, "Automatic f unctional test generation using the extended finit e state machine model", in Proc. Design Automation

al vector generationfor HDL models using linear pr · F. Fallar, S. Devadas, and K. Keutzer, "Function ogramming and 3-satisfiability", in Proc. Design Au tomation Conf., June 1998

81

to enumeration for FSMs with datapaths", in Proc. . Formal Methods in Computer Aided Design, Nov. 1998 を容風。これらの技術は強力であり、ハイレベル合成か ら生成される設計の検証におけるモデルチェック技術や . J. Kukula, T. Shiplo, and A. Aziz, "Implicit sta 定理証明とともに、将来の応用の可能性がある。

J. R. Burch, E. M. Clarke, D. E. Long, K. L. McM illan, and D. L. Dill, "Symbolic model checking fo r sequential circuit verification", IEEE Transacti ons on Computer-Aided Design, vol. 13, Apr. 1994

p.112-117, May 1994、で提案された。

· R. K. Brayton et al., "VIS: A system for verific ation and synthesis"; in Proc. Int. \Conf. Computer -Aided Verification, July 1996

. S. Owro, J. M. Rushby, and N. Shankar, "PVS: A p. rototype verification system", in 11th Internation d.), vol. 607 of Lecture Notes in Artificial Intell al Conference on Automated Deduction (D. Kapur, e igonce, Springer Verlag, 1992

【0010】ここ数年、非解釈関数によるシンボリック ツミュアーションに揺んへ特値杆チェックのたきの抵木

. R. Shostak, "An algorithm for reasoning about eq 的アルゴリズムの効率の改善について、いくつかの論文 が発表されている。

uality", Communications of the ACM, vol. 21, no. 7, pp. 583-585, 1978

· A. Goel, K. Sajid, II. Thou, A. Aziz, and V. Sing hal, "BDD based procedures for a theory of equalit f. Computer-Aided Verification, pp. 244 255, July 1 Int. Conf. Computer-Aided Design, pp. 2-6, Nov. 1995 · R. Jones, D. Dill, and J. Burch, "Efficient vali y with uninterpreted functions", in Proc. Int. Con dity checking for processor validation", in Proc.

ゅ する。その決定手続きは、算術資算とともに、ブール資 を参原。本発明でHいられるシンボリックシミュレーシ ョンアルゴリズムは、従来技術といくつかの共通点を有

· C. Barrott, D. Dill, and J. Lovitt, "Walidity ch y", in Proc. Formal Methods in Computer Aided Desi ecking for combinations of theories with equalit

算を含む。

hal, "MM hased procedures for a theory of equalit Goel, K. Sajid, Il. Thou, A. Aziz, and V. Sing y with uninterpreted functions", in Proc. Int. Con

gn, pp. 187-201, Nov. 1996

f. Computer-Aided Verification, pp. 244 255, July 1

を参照。また、要求に応じて、決定手続きに迫加の代数 を加えることも可能である。C. Barrott, D. Dill. and J. Levitt, "Validity checking for combinations of theories with equality", in Proc. Formal Methods in Computer Aided Design, pp. 187-201, Nov. 1996.

[0011] しかしながら、本発明で用いられるシンボ リックシミュレーションアルゴリズムは、ブール版算/ 条件をどのように扱うかにおいて従来技術とは異なる。 最も近いのはA. Gool at al. のものであるが、対応する 届りを記憶するのに必要なブックキーピングにおいて異 なる。A. Gool, K. Sajid, H. Thou, A. Aziz, and V. Singhal, 「UDD based procedures for a theory of equ ality with uninterproted functions", in Proc. Int. Conf. Computer-Aided Verification, pp. 244-265, Ju 1y 1998、を参照、

[0012] 1, 2, 2 徒来技術:スケジューリング

スケジューリングは、ハイレベル合成に基づく設計フローにおいて抜も重要なステップのうちの1つである。スケジューリングに関する全般的な情報については、・D. D. Gajski, N. D. Dutt, A. C.-H. Flu, and S. Y.-L. Lin, High-levelSynthesis: Introduction to Chip and System Dosign, Kluwer Academic Publishers, No

・6. De Bicholi, Synthesis and Optimization of Dig
を参照。タイミング指視を部分的にしか又は全く含まないとヘイピア記述からはじめて、設計のサイクルごとのピヘイピアは、スケジューリングステップ中に固定され、あ、このサブセクションでは、スケジューリングステップ中に延行されるいくつかの代表的な変換について説明する。それらの変換により検証プロセスの複雑さがどのように増大するかについてもここで説明する。

10013] 11.2.2.1 クロックナイクル境界 の様入] スケジューリングは、回路のピヘイピア配送か らスケジュールを導出するプロセスである。単純な形の スケジュールとがは、実行されるのは、ピヘイピア配 近にクロックサイクル境界、すなわちなかトを入れるこ とからなる策様だけである。IID に配送の場合、これに nd 山当する可能性のあるものの1つは、ピヘイピア監絡に いくつかの"wait until clk=1 and clk"イベント交を挿 入することである。詳細は、D. Knapp、T. Ur. D. Bacal illen, and R. Miller, "behavioral synthesis method alogy for IIDL-based specification and validation", in Proc. Besign Automation Conf. pp. 28-291, June 1995、を参照、あるサイクル境界と次のサイクル境界の 団の成算の当は自合は温度を要すため、一般に、いくつ かの条件を満たすために複数のカットを入れる。例え

の新しい値が有効になることが強明される。上記の塾の 結果として、スケジュールは、シミュレーション中に勧

った値を生成する可能性がある。

alwaysプロックのような暗然のループを含む)を切るためにカットを入れる。知られているように、ピヘイピアとスケジュールとは、サイクルごとに等値ではない。従って、等値性の概念と、等値性をチェックする技術とは、クロックサイクル境界を超えて作用する必要がある。周知のように、出力を計算するのに必要なフロックサイクル製は、異なるスレッドあるいは入力値に対しては異なる可能性がある。さらに、「データ体存性がある可能性がある)ループの存在もまた、機能の複雑さを掲

述は、"wait until clk=']' and clk'event"文を含まな 加されたクロックサイクル境界を示す。なお、x#var、y ループの最後の"wait for Ons"文まで有効ではないから 入の後に"wait until clk='1' and clk'event"文を導入 することにより、y#varへの代入が評価される前にu#var 可能性もある)ループの存在もまた、検証の複雑さを増 大させる。さらに、HDLの複雑なセマンティクス(例 えば、同号代入や並行文)のため、サイクル境界を導入 ある。これらのイベント文は、スケジューリング中に追 対してなされるすべての代入は信号代入である。VHD 1.における信号代入文のセマンティクスは、信号に代入 される値は切時に計算されるが、その代入はある後の時 ルタ遅延を導入し、先行する僣号代入文によって生成さ するという単純な変換でさえ、設計の機能を変更するこ る。この記述は、whileループと、さまざまな変数及び **眉号代入文を含むプロセスに関する。いくつかのステッ** プは、算術計算を含む。このプロセスでは、2つの"wai 借号y#varへの代入を考える(なお、このビヘイピア配 い)。右辺の式の計算は、信号山varの古い値を使用す である。しかし、スケジュールでは、uhvarへの信号代 t until clk=' I' and clk' "イベント文に注釈を付けて #var、u#var及びdx#varは信号であり、これらの変数に **刻まで有効にならないというものである。この時刻は、** は、デルタに箏しい。"wait for Ons;"文の目的は、デ 【0015】 ビヘイピア配述におけるwhileループ内の とがある。これは、次の例によって明確に例示される。 れた新しい値が有効になることを強制することである。 明示的な時刻が指定されていない場合、デフォルトで [0014] 例1:図1に示すVHDLTM記述を考え る。倡号uhvarへの先行する代入は実行されているが、

[0016] <u>[11, 2, 2, 2, 6) 高算の並べ替え</u>] 放算 の並べ替えは、ピヘイピア配逃に存在する並列性を利用 するため、及び、与えられたリソースを検大限に利用す るために、スケジューリング中に実行することが可能で ある。一般に、これは条件放算及び完全なループを並べ 替えることを含む。 最新のスケジューリング技術では、 しばしば、データフロー及びメモリアクセスの飲存性を 維持しながら、ピヘイピア配逃における荷算を任意に並 へ替える。詳細には、

ば、すべてのループ (VHDLのprocess文やVerilogの

(12)

特開2001-142937 (P2001-142937A)

77

 D. D. Gajski, N. D. Dutt, A. C.-H. Wu, and S. Y.
Lin, High-levelSynthesis: Introduction to Chip and System Design, Kluwer Academic Publishers, No rwell: MA. 1992 ・6. De Michell, Synthesis and Optimization of Dig ital Circuits, McGrav-Hill, New York, NY, 1994
を参照。商算の业べ替え中に導入される可能性のあるエラーには、データ依存性、条件制御依存性、及びメモリハザード (例えば、RAW (read-after-write)、WAW (writa-after-write)など)の違反がある。預算の並べ替えにより生じとの知識及びデータのフローの抽出を必要とする。さらに、制御及びデータのフローの抽出を必要とする。さらに、制御及びデータの依存性が契数において満たされることをチェックすること (例えば、指述問型チェック技術を規則チェック技術を用いて)が含まれることをチェックすること (例えば、Malimila Malidation は C. Chon, and K. Kuckenkar, Multi-diaensional rule checking for high-level design verification, in Proc. Int. High-level design Varidation & Test Wshp., Nov. 1997, etc.

[0017] 例2: 図2 (a) はピヘイピアC記述を示し、図2 (b) はその対応するスケジュールを示す。この例のピヘイピアは、シーケンシャルプログラムとして指定されているため、各スレッドで契行される演算について完全な順序を定義している。しかし、スケジューラは、保存の必要がある演算とうし間の依存性の解析を自動的に実行するかもしれないし、演算の順序が出力の計算にとって重要ではないときに演算を並べ替えることを選択することがある。このような並べ替えは、リゾースやクロック期間の教を最適にするために実行される可能。

[0018]以下の並べ替え操作が、このピヘイピアについてのスケジュールで実行されている。

・ピヘイピアにおいて+2及び\*1とマークされた紋算の原序は逆転されている。これは、基本プロック内の紋算の局所並べ替えの何である。この並べ替えは正しくない。その理由は、ピヘイピアにおける紋算+2と\*1の間にデータ依存性があり(+2の出力は\*1の入力である)、このデータ依存性は、図2に示すスケジュールでは嵌られているからである。

[0019] 2つのforループの実行順序はスケジュールでは遊覧されている。ピヘイピアにおいて最初に現れるループは、スケジュールでは状態22、S3、及びS4によって実現され、ピヘイピア配送の第2のforループは、スケジュールの状態S1で実現されている。この並へ替えは妥当である。その理由は、2つのループの間にデータ依存性や仮先頃位間ががないからである(これらのループに共通な唯一の変数であるルーブカウンタcountit、各ループの前にのに初期化される)。

[0020] [1, 2, 2, 3 パス/セグメントの祖 no

型」とヘイビア配近における相異なるパス (守なわち、 群算のスレッド) は、しばしば、異なるスケジューリン グの機会及び間約を提示する。従って、ピヘイビアにお いて与えられたパスを最大限に最適化するためには、ピ ヘイビアにおける残りのパスとは別個にパス (またはそ の部分) をスケジューリングする必要が生じることがあ る。これにより、スケジュールにおいてパスまたはパス セグメントの複製が生じる。パスに基づくスケジューリ ング技術は、ピヘイビアにおける単純 (無関路的あるい は非巡回 (acyclic) ) パスに対してこのような環適化 を行う。同様に、ルーブ指向スケジューリング技術は、  R. Camposano, "Path-basod schoduling for synthos is", IEEE Trans. Computer-Aided Design, vol. 10, p. 85-93, Jan. 1991

ピヘイピアにおける非単純パスに対してこのような最適

化を自動的に行う。

• S. Bhattacharya, S. Dey, and F. Brglez, "Perform ance analysis and optimization of schedules for co nditional and loop-intensive specifications", in P roc. Design Automation Conf., pp. 491-496, June 199

お田

このように、従来の検証ストラテジは、ピヘイピア及び することを検証する。さらに詳糊には、C.-T. Chen and checking functional and timing compatibility of s ynthesized designs", inProc. The International Sym posium on High-Level Synthesis, pp. 112-117, May 19 ントの複製は、検証プロセスの複雑さを増大させる。知 られているように、演算と変数の間の関係は一対一では なくなる。従って、構造同型をチェックする単純な技術 は、スケジュールとピヘイピアの等価性を証明するのに 十分ではない。 複製により、ピヘイピアに対するスケジ ュールにおける演算の数が増大するが、アヘイピア、も るいは、そのどヘイピア内の与えられたバスあるいはス スケジュールにおけるパスを列挙することである。さら に、対応するパスのそれぞれの対ごとに、このようなス トラテジは、ピヘイピア及びスケジュールにおいて実行 される演算のセットが同型のデータフローグラフを形成 A. Parker, "A hybrid numeric/symbolic program for 【0021】また、スケジューリング中のパス/セグメ レッドに沿って実行される演算のセットは同一である。

[0022] [1, 2, 2, 4 ルーブ変換] ルーブ は、しばしば、ピヘイピア配近においてパフォーマンス あるいはパワーに関するクリティカルな部分を構成す る。データ独立ループ(実行回数が専前に既知であり、 入力値とは独立なループ)、及び、データ仮将ループ (実行回数が静めに既知ではなく、入力データに依存す るループ)を積極的に厳密化するきまざまなスケジュー リング技術が提案されている。これらの技術には以下の

におけるループの境界は、ピヘイピアにおける対応する y)のいくつかのコピーに変換した後、そのループのコピ 一をすることである。 第2の意味は、スケジュールにお けるルーブの1回の実行が、ピヘイピアにおけるループ の複数回の実行に対応することである。 2 種類のループ 【0024】・ループ回転。これにより、スケジュール ルーブの境界に対してずれる。ループ回転を図3 (d) は、ビヘイピアにおけるループをループ木体(loop bod ・ループ展団。 ケーブ展団の1つの荷味 原朋変換を図3 (b)及び図3 (c)に倒示する。 0023

折骨み(loop folding)あるいはルーブ巻付け(loop wind ing)ともいい、ループ本体の複数回の実行を並行して実 行するものである。これには、正当性を保証するために プロローグ及びエピローグを作成することも必要になる Nicolau, and D. Gajski, "Percolation based synthe sis", in Proc. Design Automation Conf., pp.444-44 【0025】・ループパイプライン化。これは、ループ ことがある。さらに許細には、R. Potasman, J. Lis,A. 9, June 1990, を参照。ループパイプライン化を図3 (e) に例示する。

能である場合、あるいは、ループ実行回数が一定で既知 スの列挙は至難となる。さらに、回転やパイプライン化 るスレッドあるいはパスの列挙は、ループの相異なる実 行される回数はデーク放存であることがあり、静的に限 である場合であっても、ピヘイピア及びスケジュールに おける異なるパスの個数により、すべてのこのようなパ のようなループ最適化は、スケジュールとピヘイピアに **複雑にする。特に、ピヘイピア及びスケジュールにおけ** 行カウントを考慮する必要がある。 さらに、ループが実 定することが困難である。さらに、このような限定が可 おけるループの境界どうしの間の対応を破壊する。本発 明の重要な特徴は、スケジュールにおけるすべての非単 [0026] ピヘイピアにおけるループの存在と、スケ ジューリング中のループ最適化の適用は、検証を非常に 純パスの列挙を避ける、ループ不変項の自動抽出にあ

[0027] [1, 2, 2, 5 段機実行] 投機実行で は、ビヘイピア記述の一部が、その部分を実行すること 、イレベル合成のスケジューリングステップに統合され **牧機実行によって、検証は更に複雑になる。 角要な点で** あるが、ピヘイピアでの制御依存性は、投機実行を含む が必要であるとわかる前に、実行される。投機実行は、 ると、大幅なパフォーマンス改善が得られる。しかし、 スケジュールでは満たされない。さらに詳細には、

· 1. Radivojevic and F. Brewer, "Ensemble represen scheduling", in Proc. High-level SynthesisWorksho tation and techniques for exact control-dependent pp. 60-65, 1994

· O. Lakshminarayana, A. Raghunathan, and N. K. Jh a, "Incorporating speculative execution into sched uling for control-flow intensive behaviors", in Pr oc. Design Automation Conf., pp.108-113, June 1998

【0028】 スケジューラは、投機実行される演算の結 果を格納するためにスケジュールに追加一時変数を導入 するのが一般的である。また、スケジューラは、それら の一時変数が依存する投機条件が評価された後にそれら の一時変数を解決するための追加コード(代入文)を生 成する。構造同型に基づく検証技術は、このような変換 を倹証することができない。これについては、

· J. Gong, C. T. Chen, and K. Kucukcakar, "Multi-d imensional rule checking for high-level design ver ification", in Proc. Int. High-level Design Valida

· C.-T. Chen and A. Parker, "A hybrid numeric/symb olic program for checking functional and timing co mpatibility of synthesized design", in Proc. The I nternational Symposium on High Level Synthesis, p tion & Test Wkshp., Nov. 1997 p. 112-117, May1994

に説明されている。

[0029]

【発明が解決しようとする課題】 [2. 発明の概要] 本 発明は、新規な非解釈シンボリックシミュレーション手 **続きに関する。本発明の技術は、ピヘイピア仕様及びス** ケジューリングされたRTLが与えられた場合に、2つ の配述の出力が相互に無条件に対応するかどうかを判定

借号の間の条件付き信号対応を出力へ向かって伝搬させ 合に、相互に対応する。その場合、出力が対応するため イビア記述におけるループと、スケジューリング中のル 一プ変換との存在によって、非常に複雑になる。本発明 ルーブ変換とが存在する場合に、等価性チェッカによっ [0030] スケジューリングされたRTLとビヘイビ ア記述の間の、条件付きの可能性がある入力対応のリス トからはじめて、木発明の技術は、2つの配述における る。2つの資算の出力は、その資算型が同一であり、か つ、ある条件下でその演算への入力が相互に対応する場 【0031】算術演算とは異なり、ブール演算は完全に 解釈される。これにより、条件を超えて前算を移動させ るような変換の正当性のチェックが可能となる。このよ 【0032】スケジューリングを検証する作業は、ピヘ の重要な特徴は、ループと、スケジューリングにおける て、スケジュールとどヘイピアにおける信号の間の対応 形式で、不変項の効率的な抽出を行うことである。本発 の条件は、入力が対応するための条件の論理額となる。 うな変換は、スケジューリングにおいて一般的である。

制御状態よりもデータパスレジスタによって引き起こさ **明の技術は、ほとんどの散計における状態空間爆発は、** 

特間2001-142937 (P2001-142937A)

た場合に、 脱った否定 (フォールスネガティブ) を報告 ついては、その応用の具体例とともに、セクション4で リングにおける代表的な変換とみなされるものに基づい **れるほとんどの設計を検証することが可能である。本発** するという点で、悲風的である。本発明の技術の詳細に れるとこう概要に、部分的に基ろいている。メケジュー て、本発明の技術は、スケジューリングによって生成さ 明の技術は、扱うことができないルーブ最適化に遭遇し

【0033】シンポリックシミュレーションアルゴリズ 4は、本発明の重要な構成要装であるが、本発明の主要 な貢献ではない。本発明の主要な貢献は、ループを扱う ことが可能な、無閉路グラフに対する基本的なシンポリ ックシミュレーションアルゴリズムの改善にある。

れたスケジュールでも使用可能である。なお、従来の技 は、他の最適化技術を適用したスケジュールにも適用可 [0034] 従来の方法における問題点を解決するため 記述との等価性を証明する改善された方法を提供するこ とである。本発明は、いかなるスケジュールにも削限さ れず、従来の技術の項で説明したいかなる最適化がなさ に、木箔明の目的は、スケジュールと、そのピヘイピア 筋の項で説明した最適化は単なる例示であり、本発明

フローグラフあるいは制御/データフローグラフ (CD 【0035】ピヘイピアは、従来の任意の形式で指定す ることができる。これには、制御フローグラフ、データ FG:control/data flow graph) 及びピヘイピア

y, D. MacMillon, and R. Willer, "Behavioral synthes (超) 状態マシンが含まれるが、これらには限定されな い。 ピヘイピア合成についての詳細は、D. Knapp, T. L is methodology for HDL based specification and val idation", in Proc. Design Automation Conf., pp. 28-291, June 1995、を参照。

【0036】本発明は、ピヘイピア及びスケジュールに おけるプライマリ入力変数の間の対応が与えられている こと、及び、出力変数間の対応と、出力変数が同一の値 を有すると期待される時刻とが明確に指定されているこ とを仮定する。本発明は、複数のループ、ネストしたル **ープ、及びデータ依存ループを含むピヘイピア及びスケ** ジュールを処理する。

【0037】本発明の検証手続きの正確さ及び完全性を 保証するために設計及び合成のフローが満たすことが必 【0038】・ピヘイピア配近における演算は、スケジ **熨とされる仮定は以下の通りである。** 

プロセス中には、そのゲートレベル実装に分解されない。 について、n回の反復後の停止を考え、すべての場合に 扱われるもの (例えば、算術及び比較預算) と、分解ま 算)とに分けることができる。例えば、ワードあるいは ューリングプロセス中にアトミックエンティティとして ビットベクタ商算(例えば加算)は、スケジューリング たは変換される可能性のあるもの (例えば、ブール荷

算を非解釈のまま残すべきかを決定するために使用され 。 ことがある。アトミック商勇と非アトミック商勇に商勇 を分けることは任意性を伴うことがあるが、検証手続き に与えられることが必要である。この情報は、本発明の 検証技術の主要な情成要素である非解釈シンポリックシ ミュレーション手続きにより、どの資類を解釈しどの演

ば、算術及び比較演算がアトミックであると宣言される 場合、スケジューリングは、スケジュールを最適化する ために、これらの荷算の機能についての知識を使用しな い。比較預算は、分岐及びループ終了条件を決定するた [0039]・スケジューリングプロセスは、アトミッ ヶ済算の解釈から導き出される知識を使用しない。例え めに使用されるものを含む。

てエラーありとしてフラグが立てられる。なお、この仮 定は、ループ本体あるいは境界がピヘイピアとスケジュ ろ、これは、ループ原間がピヘイピアからスケジュール [0040]・ピヘイピアにおける各ループごとに、ス り、スケジュールにおけるループの1回の実行は、ビヘ イビアにおけるループの1回以上の実行に対応する。こ の性質を満たさないスケジュールは、検証手続きによっ ールとで同一であることを要求するものではない。 むし へと実行されているだけであり、その逆ではないことを ケジュールには少なくとも1つの対応するルーブがあ 意味する。

【0041】 七記の仮定はそれほど制限的ではない。そ の理由は、これらの仮定は、リストスケジューリング、

R)、パスに基づくスケジューリング、ループ指向スケジ ューリング(loop-directed scheduling)などのような周 知のスケジューリングアルゴリズムを含む最も実際的な スケジューリング技術によって満たされるからである。 強御的スケジューリング(force-directed schedulin これについては、

and System Design, Kluwer Academic Publishers, No -L. Lin, High-levelSynthesis: Introduction to Chip . D. D. Gajski, N. D. Dutt, A. C.-II. Wu, and S. Y. rwell, MA, 1992

· G. De Micheli, Synthesis and Optimization of Dig ital Circuits, McGraw Hill, New York, NY, 1994 に示されている。

【0042】本明細番では、代表的なスケジューリング 技術という用語は、上配の仮定を満たすスケジューリン グのアルゴリズムあるいはツールを表すために使用す

合、木発明のアプローチはループ不変項を使用するもの 木体の後のコードが実際に実行されるかどうかには特に 対処しない。ある意味で、本発明では、すべてのn≥0 である。しかし、ループ停止の周盥、すなわち、ループ 【0043】ループについて正当性をチェックする場

3

等価性をチェックする。本発明のアプローチのこの特徴 は強闘しなければならない。すなわち、火<u>一乙本体の工 へての反似回数に対する挙価性がチェックされる。</u>な お、算術衛貸を扱うために非解釈関数を使用するため、 解釈された館(これがスケジューラによって利用された かどうかにかかわらず)に依有する停止条件を考慮する ことはこのフレームワークでは不可能である。例えば、 ルーブが6回実行される場合に限りエラーが生じるが、 核 f 条件のために、ループは2回より多くは狭して実行 されないとする。この場合、本発明の手数きは、フォー 発明は、2回の実行後の時止のみならず、n=6を含むすべての回接nの後の停止を考慮するからである。ループ反復回接が一定の上限を4するような場合を早期停止 (early termination)という。 [ 原型を解決するための手段] 本義明の目的を造成するため、回路のスケジューリングの正当性をチェックする方法が提供される。回路に対するスケジュールは、ピヘイビア起送から得られる。この方はは、ルーブが回路内にあるときに非巡回スレッドの十分なセットを決定するためにルーブ不変項を抽出するステップと、ルーブ不要な他出するステップと、非巡回スレッドの等価性を証明するステップと、非巡回スレッドの等価性を証明するステップとを有する。

[0045] 好ましくは、ピヘイピア記述は、サイクル境界の導入によって要換される。

【0046】好ましくは、ピペイピア配近は、荷算並べ 替えによって変換される。

[0047] 好ましぐは、ピヘイピア記述は、ループの 原開、巻付け、近畳み及びパイプライン化によって変換

[0048] 好ましくは、ピヘイピア配近は、荷算の投版実行によって変換される。

【0049】本語明のもう1つの特徴によれば、国路の ピヘイピア記述に対して回路のスケジュールを検証する 方法が提供される。この方法は、前記スケジュールか ら、ループを含む可能性のある実行のスケジュールスレッドを選択するステップと、前記ピヘイピア監証から対 応するピヘイピアスレッドを類別するステップと、スケジュールスレジュールスレンドをピヘイピアンメレッドを難別するステップと、スケジュールスレンドとピヘイピアといかドの無条件等値性 を選出するステップと、実行のすべてのスレッドについて繰り返すステップと、実行のすべてのスレッドについ

【0051】 がましくは、ピヘイピアは、ピヘイピア状 随道はグラフとして指定される。

【0052】好ましくは、前記証明するステップは、前 虹スケジュールスレッドをスケジュール掃造グランに変 執するとともに前記ピヘイピアスレッドをピヘイピア将

よっ 治グラフに変換するステップと、前部スケジュール構造 グラフと前記ピペイピア構造グラフの等面性をチェック するステップとを行する。 【0053】本発明のもう1つの特徴によれば、回路の ピヘイビア記述に対して回路のスケジュールを検証する 方法が提供される。この方法は、スケジュールをスケジュール状態面移グラフとして指定するステップと、回路のピーイビアをピペイビアをは各グラフとして表現するステップと、回路のドーイビアをピペイビアが面積グラフから、対応するピーイビアに関ロにイエアスレッドを確別するステップと、前記スケジュールスレッドを確別するステップと、前記スケジュールスレッドをなアップと、前記スケジュールはピアスレッドをピールで開始グラフルで表現するステップと、前記スケジュール研究がフレに変換するともに前記ステップと、前記スケジュール研究グラフに変換するアップと、前記スケジュール研究グラフに変換するアップと、対策値グラフの等価性をチェックするステップと、実行のすべてのスレッドについて繰り返すステップと、実行のすべてのスレッドについて繰り返すステップとを有する。

ルスネガティブを報告することになる。その理由は、本

2をたどり、a r r 2内の各ノードを処理して、スケジ ュール構造グラフの入力からスケジュール構造グラフの ピア構造グラフ内の信号の識別子であり、cは等価性の かをチェックするステップと、arr2内のすべての出 一ドが等価であることがわかった場合に等価性を見つけ 前記ピヘイピア構造グラフ内のすべてのノードを含む脱 序セットa r r 1を作成するステップと、前記ピヘイビ ジュール構造グラフ内のすべてのノードを含む順序セッ と、ピヘイピア構造グラフ内の非甚低変数を基底変数で **扱すステップと、スケジュール悄潰グラフ内の入力ノー** ドに対する等価性リストを構成するステップと、BFF 川力へ等価性リストを伝収させるステップと、各等価性 ピア群治グラフ内の対応する出力ノードで移信有が臨床 ライマリ川カノードに対するトートロジーであるかどう 【0054】好ましくは、等価性チェックは、前配ピヘ イビア状態選移グラフ内の各ノードが祓ノードの推移フ ア構造グラフ内の各ノードが抜ノードの推移ファンイン 内のすべてのノードの後にのみ現れるように、前配スケ ピヘイピア構造グラフ内の基底変数を識別するステップ リストMのエントリロ丼 (n, c) であり、uはどへイ 条件を投す二分決定ダイヤグラムであるとして、ピヘイ したかどうか、及び、対応する条件にがarr2内のブ カノードについて繰り返すステップと、すべての山カノ トロェに2を作成するステップと、ロドェ1をたどり、 アンイン内のすべてのノードの後にのみ現れるように、 たとするステップとを有するプロセスによって行われ

[0055]本発明のもう1つの特徴によれば、回路のスケジュールと該回路のピペイピアとの間の等値性を検証する力法が提供される。前記スケジュール及び前記ピペイピアは、実行の巡回スレッドを有する可能性がある。前記方法は、スケジュールをスケジュール状態強移

(16)

特開2001-142937 (P2001-1429374)

29 グラフとして表現するステップと、ピヘイピアをピヘイ ピア状態強移グラフとして接現するステップと、前記スケジュール状態逸移グラフやの強連結成分を確別するス デップと、各強連結成分内の終了ノードを確別するステップと、前記スケジュール状態逸移グラフをつぶして、前記法は成分を確別するステップと、前記法は分を通らないサブバスを併合するステップ

ての状態遷移決定をカプセル化するパスシグナルを生成 るステップと、強択された対応セットが、前のシンボリ であるかどうかをテストするステップと、前配出力等価 と、終了点が高々3度現れるようにルートからシンクへ と、パスシグナルを用いて、関約されたシンボリックシ ミュレーションを実行してピヘイピア状態遺移グラフ内 の対応するパスを織別し、簸パスに対する構造RTL回 路を取得するステップと、選択されたパスにおいて、以 と、選択されたパス内の選択された強連結成分に対する ックシミュレーションの強連結成分カットにおいて得ら れる変数対応より小さい場合に、シンボリックシミュレ 性が条件付きである場合に非等価性を報告してこの方法 を終了するステップと、遊択されたパス内のすべての強 と、選択されたパスに対する構造RTL回路を取得する ステップと、選択されたパスを列挙するのに必要なすべ 不変項を、対応セットのリストとして抽出するステップ ーションを再実行するステップと、対応セットのリスト 内の各対応セットについて以上のステップを繰り返すス テップと、出力等価性条件が、パス条件以外の条件付き と、対応セットのリストから1つの対応セットを選択す するための国路を構造RTL回路に追加するステップ 通結成分について以上のステップを繰り返すステップ と、以前に選択されていないパスを選択するステップ 前に選択されていない強連結成分を選択するステップ

【0056】好ましくは、関約されないシンボリックシミュレーションが、ピヘイビア状態選移グラフの始状態を解释パスリストに割り当てるステップと、群歩パスリスト内で以前に訪れていない状態を選択するステップと、非解釈シンボリックシミュレーションを実行して、スケジュール保造RTL及びピヘイア保造RTL内の対応する信号を確別するステップと、道移条件とパスングナルの協理積がゼロでない場合に、状態S」の新しいコピーを替びとに前記値加するステップと、SiからS」への各出選移がスにに前記がするステップと、SiからSiへの各出資格がスにに前記があるアップを繰り返すステップと、対象ステップと、対象のあが表状態のインなクスとなるまで、すべての訪れていない状態のインスタンスとなるまで、すべての訪れていない状態について繰り返すステップとを有するプロセスを用いて実行さ

【のの51】好ましくは、不変項は、各ループごとに、 各カットが削配ループの各実行の境界における姿骸値を 妻すような、スケジュール内のパスの構造RTL回路内

カットにおける等価関係を、等価関係セットの集合に追 けるパスの構造RTL回路内の対応するカットを織別し るステップと、スケジュール及びピヘイピアのRTL凶 路における対応するカットの各対における変数どうしの 間の等価関係を識別するステップと、域後のカットと域 をチェックするステップと、前記関係が同一でなく、か つ、最後のカットにおける幹価関係が、最後の前のカッ トにおける箏伽関係のサブセットである場合、最後の前 **実行について2つのRTL回路を展開して、繰り返すス** テップと、前記場係が同一でなく、かつ、吸後のカット における等価関係が、最後の前のカットにおける等価関 のゲープ状行について2つのR.T.L回路を展出して、禁 加するステップと、等価関係セットの集台内で、他のエ ントリのスーパーセットであるすべてのエントリを削除 するステップと、等価関係セットの最終集合を、不変項 の所留の集合として指定するステップとを有するプロセ アヘイアアにお て、第1と第2のカットの間のサブ回路と、第2と第3 のカットの間のサブ回路が同型であることをチェックす 後の前のカットとの声の等価関係が同一であるかどうか のカットにおける等価関係を破棄し、1の以上のループ 係のサブセットでない場合、最後の前のカットにおける **尊価関係を、尊値関係セットの集合に追加し、1つ以上** り返すステップと、前記関係が同一である場合、最後の の3個のカットを識別するステップと、 スを用いて、ルーブから抽出される。

[0058]本途男のもう1つの特徴によれば、回路のスケジューリングの正当在をチェックするシステムが提供される。回路に対するメケジュールは、ピヘイピア社送から得られる。このシステムは、ルーブが存在するときに非巡回スレッドの十分なセットを改定するルーブ不変項が出器と、値割ルーブ不変項を抽出するシンボリックシミュレータと、非巡回スレッドの等値柱を証明する

のすべてのパスについて以上のステップを繰り返すステ

等価性証明器とを有する。 【0059】 好ましくは、前配ピヘイピア韶楽は、サイクル境界の導入によって変換される。 【0060】 印ましくは、通知アヘイアア記述は、 密算型 く替えによって複数される。

ゴーロストナン、文字の175。 【0061】 好ましくは、何昭ピヘイピア記述は、ルーブの原間、巻付け、佐畳み及びパイプライン化によって変換される。 【0062】 母ましくは、前的アヘイアア慰迦は、逆算の牧職実行によって狡骸さむる。

[0063]本発明のもう1つの特徴によれば、回路のピヘイピア配近に対して回路のスケジュールを後請するシステムは、スケジュールをなオケジュール状態適移グラフとして指定するスケジュール状態適移グラフシェネレータと、回路のピヘイピアをピヘイピア状態適移グラフシェネレータと、同路のピヘイピアない電路をグラフジェネレータと、哨記スケジュール状態 過移グラフから、実行のスケジュールスレッドを選択する

るスケジュールスレッドセレクタと、前配どヘイビア状 スレッドをスケジュール構造グラフに変換するともに するコンパータと、前記スケジュール構造グラフと前記 **側灘移グラフから、対応するピヘイピアスレッドを選択** するピヘイピアスレッドセレクタと、前配スケジュール **前記とヘイピアスレッドをピヘイピア構造グラフに変換** ピペイピア情治グラフの等価性をチェックする等価性チ エッカとを有する。

[0064] 本発明のもう1つの特徴によれば、回路の スケジューリングの正当性をチェックするための、プロ ムが前記チェックを実行することを可能にする命令を含 み、抜命合は、ループが存在するときに非巡回スレッド セッサ及びメモリを有するコンピュータシステムが提供 される。回路に対するスケジュールは、ピヘイピア配近 から得られる。前記メモリは、前記コンピュータシステ の十分なセットを決定するためにループ不変項を抽出す **シミュレーションの命令と、非巡回スレッドの等価権を** る命令と、ループ不変項を抽出するためのシンポリック 証明する命令とを含む。

1.ろ命令を含む。

【0065】好ましくは、何點とヘイピア配当は、サイ クル境界の導入によって変換される。 【0066】 外ましくは、前記ピヘイピア記述は、演算 並べ替えによって変換される。

ブの原明、巻付け、折畳み及びパイプライン化によって [0067] 好ましくは、包記アヘイピア記説は、ルー 変換される。 【0068】 好ましくは、前配ピペイピア配法は、液算 の投機実行によって変換される。 [0069] 本苑明のもう1つの特徴によれば、回路の ピヘイピア記述に対して回路のスケジュールを検証する ための、プロセッサ及びメモリを有するコンピュータシ 含み、豚命令は、スケジュールをスケジュール状態遷移 グラフとして指定する命令と、回路のピヘイピアをピヘ イビア状態遷移グラフとして表現する命令と、前配スケ ジュール状態遷移グラフから、実行のスケジュールスレ ッドを選択する命令と、前記ピヘイピア状態遷移グラフ **新記スケジュールスレッドをスケジュール構造グラフに** ラフと前配ピヘイピア構造グラフの等価性をチェックす システムが前配検証を実行することを可能にする命令を ステムが提供される。前記メモリは、前記コンピュータ 構造グラフに変換する命令と、前配スケジュール構造グ 変換するとともに値配とヘイピアスレッドをピヘイピア から、対応するどへイビアスレッドを遊択する命令と、

【0070】本発明のもう1つの特徴によれば、回路の アヘイピア記述に対して回路のスケジュールを検託する ための、プロセッサ及びメモリを有するコンピュータシ システムが、スケジュールをスケジュール状態遷移グラ ステムが姫供される。前記メモリは、前記コンピュータ

る命令と、実行のすべてのスレッドについて繰り返す命

スレッドを選択するステップと、前配ピヘイピア状態圏 **構造グラフに変換するとともに前配ピヘイピアスレッド** イビア状態選移グラフとして表現するステップと、前記 スケジュール状態遷移グラフから、実行のスケジュール ケジュール構造グラフと前配ピヘイピア構造グラフの等 移グラフから、対応するピヘイピアスレッドを類別する ステップと、前配スケジュールスレッドをスケジュール をピヘイピア構造グラフに変換するステップと、前配ス 価性をチェックするステップと、実行のすべてのスレッ フとして指定するステップと、回路のピヘイピアをピヘ ドについて繰り返すステップとを実行することを可能に

ードが抜ノードの推移ファンイン内のすべてのノードの 後にのみ現れるように、前配ピヘイピア構造グラフ内の 変数を識別するステップと、ピヘイピア構造グラフ内の ケジュール構造グラフの出力へ等価性リストを伝搬させ すべてのノードを含む順序セットarrlを作成するス テップと、前記どヘイピア構造グラフ内の各ノードが鼓 と、arr1をたどり、ピヘイピア構造グラフ内の基底 非甚底変数を甚底変数で丧すステップと、スケジュール 構造グラフ内の入力ノードに対する等価性リストを構成 ードを処理して、スケジュール構造グラフの入力からス 【0071】好ましくは、前配命令は、前配コンピュー タシステムが、前記ピヘイピア状態遷移グラフ内の各ノ ノードの推移ファンイン内のすべてのノードの後にのみ 現れるように、前配スケジュール構造グラフ内のすべて のノードを含む順序セットarr2を作成するステップ するステップと、arr2をたどり、arr2内の各ノ るステップと、各等価性リスト内のエントリは対 (u.

c)でやり、uはアヘイアア構造グラフ内の信号の観別 ムであるとして、ピヘイピア構造グラフ内の対応する出 カノードで等価性が確定したかどうか、及び、対応する ップと、すべての川カノードが等価であることがわかっ 子であり、 c は等価性の条件を要す二分決定ダイヤグラ 条件でがarr2内のプライマリ出力ノードに対するト た場合に等価性を見つけたとするステップとを実行する a r r 2内のすべての出力ノードについて繰り返すステ ートロジーであるかどうかをチェックするステップと、 ことを可能にする命令をさらに含む。 【0072】本苑明のもう1つの特徴によれば、回路の スケジュールと駁回路のどヘイピアとの間の等価性を検 **ータシステムが提供される。前記スケジュール及び前記** ラフ内の強連結成分を職別するステップと、各強連結成 証するための、プロセッサ及びメモリを有するコンピュ ピヘイピアは、実行の巡回スレッドを有する可能性があ 5。前記メモリは、前記コンピュータシステムが、スケ ジュールをスケジュール状態遷移グラフとして表現する ステップと、ピヘイピアをピヘイピア状態遺移グラフと して安現するステップと、前記スケジュール状態遷移グ 分内の終了ノードを織別するステップと、前記スケジュ

(18)

特 閉2001-142937 (P2001-142937A)

出力等価性条件が、非等価性を報告するパス条件以外の 条件付きであるかどうかをテストし、前記出力等価性が **ール状態選移グラフをつぶして、前記強連結成分を通ら** れていない強連結成分を選択するステップと、選択され たパス内の選択された強連結成分に対する不変項を、対 再実行するステップと、対応セットのリスト内の各対応 選択されたパス内のすべての強連結成分について以上の ステップを繰り返すステップと、終了点が高々 3 度現れ ないサブパスを併合するステップと、以前に選択されて いないパスを選択するステップと、選択されたパスに対 する構造RTL回路を取得するステップと、選択された パスを列挙するのに必要なすべての状態通移決定をカブ セル化するパスシグナルを生成するための回路を構造R て、制約されたシンボリックシミュレーションを実行し るステップと、選択されたパスにおいて、以前に選択さ 応セットのリストとして抽出するステップと、対応セッ と、選択された対応セットが、前のシンボリックシミュ レーションの強連結成分カットにおいて得られる変数対 応より小さい場合に、シンポリックシミュレーションを てピヘイビア状態選移グラフ内の対応するパスを識別す 条件付きである場合にこの検証を終了するステップと、 TL回路に追加するステップと、パスシグナルを用い セットについて以上のステップを繰り返すステップと、 トのリストから1つの対応セットを選択するステップ

容パスリストに削り当てるステップと、酢容パスリスト に前記追加するステップを繰り返すステップと、許容パ 内で以前に訪れていない状態を選択するステップと、ビ 【0073】好ましくは、前配命合は、前配コンピュー ポリックシミュレーションを実行して、スケジュール構 識別するステップと、遷移条件とパスシグナルの輪県積 スとなるまで、すべての訪れていない状態について繰り タシステムが、ピヘイピア状態遷移グラフの始状態を許 ヘイピア構造RTLを生成するステップと、非解釈シン 造RTL及びどへイビア構造RTL内の対応する何号を スに追加するステップと、SjからSjへの各出遷移ごと ス内に残る訪れていない状態のみが終状態のインスタン 返すステップとを実行することを可能にする命令をさら がゼロでない場合に、状態Sjの新しいコピーを桁容パ

路が同型であることをチェックするステップと、スケジ **一ル内のパスの構造RTL回路内の3個のカットを観別** 回路内の対応するカットを織別して、第1と第2のカッ 【0074】好ましくは、前配命合は、前配コンピュー タシステムが、各ループごとに、各カットが前點ループ の各実行の境界における変数値を表すような、スケジュ するステップと、ピヘイピアにおけるパスの構造RTL トの国のサブ回路と、第2と第3のカットの国のサブ回

ける等価関係が、最後の前のカットにおける等価関係の。 サブセットである場合、最後の前のカットにおける等価 ュール及びどへイピアのRTL回路における対応するカ るステップと、最後のカットと豊後の前のカットとの間 プと、前記関係が同一でなく、かつ、尿後のカットにお トであるすべてのエントリを削除するステップと、等価 ットの各対における変数どうしの間の等価関係を説別す TL回路を展開して、繰り返すステップと、前起関係が 最後の前のカットにおける等価関係のサブセットでない 場合、最後の前のカットにおける等価関係を、等価関係 2つのRTL回路を展開して、繰り返すステップと、前 **起関係が同一である場合、最後のカットにおける等価関** 係を、等価陽係セットの集合に追加するステップと、等 価関係セットの集合内で、他のエントリのスーパーセッ 関係セットの最終集合を、不変項の所望の集合として指 の等価関係が同一であるかどうかをチェックするステッ 関係を破棄し、1つ以上のループ実行について2つのR セットの集合に追加し、1つ以上のループ実行について 同一でなく、かつ、最後のカットにおける等価関係が、

【0075】本発明のもう1つの特徴によれば、コンピ ュータが回路のスケジューリングの近当性をチェックす **一タ可能媒体を有するコンピュータプログラム製品が提** 供される。回路に対するスケジュールは、ピヘイピア記 祈から飾られる。 粒節コンピュータコードは、ループが 存在するときに非巡问スレッドの十分なセットを決定す ることを可能にするコンピュータコードを合むコンピュ と、ループ不変項を抽出するためのシンボリックシミュ るためにループ不変項を抽出するコンピュータコード

> るようにルートからシンクへのすべてのパスについて以 上のステップを繰り返すステップとを用いて前記検証を

実行することを可能にする。

**ドするステップとを実行することを可能にする命令をさ** 

レーションのコンピュータコードと、非巡回スレッドの [0076] 好ましくは、信動アヘイピア制治は、サイ 等価性を証明するコンピュータコードとを合む。 クル境界の導入によって変換される。

[0077] 好ましくは、煎記ピヘイピア記述は、没算 並べ替えによって変換される。

プの展開、巻付け、折畳み及びパイプライン化によって [0078] 好ましくは、煎沼アヘイパア記消は、ルー 変換される。

[0079] 好ましくは、前記ピヘイピア記述は、後類 の投機実行によって変換される。 [0080] 本発明のもう1つの特徴によれば、コンピ 配コンピュータが、スケジュールをスケジュール状態圏 移グラフとして指定することを可能にするスケジュール ュータが回路のどへイピア記述に対して回路のスケジュ **一ルを徐庇することを可能にするコンピュータコードを 含むコンピュータ可説媒体を有するコンピュータプログ** ラム製品が提供される。前記コンピュータコードは、前 状態遷移グラフジェネレータコードと、前配コンピュー

タが、回路のピヘイピアをピヘイピア状態強移グラフと

8

して指述することを可能にするピヘイビア状態菌移グブンジェキレータコードと、軸沿コンピュータが、超沿メケジュール状態強移グラフから、実行のスケジュールスレッドを選択することを可能にするスケジュールスレッドセンタコードと、軸記コンピュータが、耐況ピヘイビアスレッドをンドをことを可能にするピヘイビアスレッドをとクテランから、対応するピヘイビアスレッドをレクタコードと、軸記コンピュータが、軸沿スケジュールスレッドをスケジュール接着グランに接続するとともに耐温ピヘイビアスレッドをピークが、軸沿スケジュールスロードと、右記コンピュータが、画記スケジュールスロードと、右記コンピュータが、正式スケジュール構造グランに接続グランに登録にするコンパータコードと、前記コンピュータが、正式スケジュール構造グランと回路にするコンパータコードと、前記コンピュータが、正式スケジュール構造グランと回路にするにからかが、正式スケジュール構造グランと回路にするコンパーを含むない。

グラフの等価性をチェックするステップと、実行のすべ 【0081】本発明のもう1つの特徴によれば、コンピ **一ルを検証することを可能にするコンピュータコードを** ラム製品が提供される。前記コンピュータコードは、前 **記コンピュータが、スケジュールをスケジュール状態選** 移グラフとして指定するステップと、回路のどへイビア と、前むスケジュール状態選移グラフから、実行のスケ ジュールスレッドを強択するステップと、前部ピヘイピ ア状態遷移グラフから、対応するピヘイピアスレッドを 織切するステップと、前記スケジュールスレッドをスケ ジュール構造 グラフに変換するとともに前記ピヘイピア と、前記スケジュール構造グラフと前記ピヘイピア構造 てのスレッドについて繰り返すステップとを実行するこ ュータが回路のどへイビア記述に対して回路のスケジュ 会むコンピュータ可説媒体を省するコンピュータプログ をピヘイピア状態遺移グラフとして表現するステップ スレッドをどへイビア構造グラフに変換するステップ とを可能にする。

(n, c)であり、uはどヘイビア構造グラフ内の信号 BD にのみ現れるように、前記スケジュール構造グラフ内の テップと、urr1をたどり、ビヘイピア構造グラフ内 を構成するステップと、Brr2をたどり、Brr2圴 からスケジュール構造グラフの川力へ等価性リストを伝 前記コンピュータが、前記ピヘイピア状態遺移グラフ内 一ドの後にのみ現れるように、前配とヘイビア構造グラ フ内のすべてのノードを含む船所セットロロロ1を作成 するステップと、前記ピヘイピア構造グラフ内の各ノー ドが波ノードの推移ファンイン内のすべてのノードの後 すべてのノードを含む順序セット8552を作成するス の基底変数を織別するステップと、ピヘイピア構造グラ フ内の非当底変数を基底変数で表すステップと、スケジ ュール構造グラフ内の入力ノードに対する等価性リスト の各ノードを処型して、スケジュール構造グラフの入力 做させるステップと、各等価倍リスト内のエントリは対 の各ノードが該ノードの推移ファンイン内のすべてのノ 【0082】好ましくは、道語コンピュータコードは、

36 の識別子であり、cは等価性の条件を数す二分状定ダイ セグラムであるとして、ピヘイピア構造グラフ内の対応 する川ルノードで等価性が確定したかどうか、及び、対 広する条件。がarr2内のプライマリ出力ノードに対 するトートロジーであるかどうかをチェックするステッ プと、arr2内のチィでの出力ソードについて繰り返 オステップと、すべての出カソードについて繰り返 オステップと、すべての出カソードが等値であることが わかった場合に等値性を見つけたとするステップとを残 行することを可能にする。

**我現するステップと、ピヘイピアをピヘイピア状態選移** を構造RTL回路に追加するステップと、パスシグナル るステップと、対応セットのリスト内の各対応セットに ついて以上のステップを繰り返すステップと、出力等価 性条件が、パス条件以外の条件付きであるかどうかをテ Fのステップを繰り返すステップと、終了点が高々 3度 見れるようにルートからシンクへのすべてのパスについ ュータが回路のスケジュールと核回路のどへイビアとの 間の等価性を検託することを可能にするコンピュータコ 強連結成分内の終了ノードを臨別するステップと、前記 スケジュール状態選移グラフをつぶして、前配強運結成 分を通らないサブパスを併合するステップと、以前に謝 パスに対する構造RTL回路を取得するステップと、選 状されたパスを列挙するのに必要なすべての状態遷移決 定をカプセル化するパスシグナルを生成するための回路 を用いて、制約されたシンポリックシミュレーションを 以行してピヘイピア状態強移グラフ内の対応するパスを **識別し、抜パスに対する構造RTL回路を取得するステ** ップと、選択されたパスにおいて、以前に選択されてい ない強連結成分を選択するステップと、選択されたパス 内の選択された強連結成分に対する不変項を、対応セッ トのリストとして抽出するステップと、対応セットのリ ストから 1 つの対応セットを選択するステップと、選択 された対応セットが、前のシンポリックシミュレーショ ンの強速結成分カットにおいて得られる変数対応より小 ストするステップと、前記川力等価性が条件付きである 場合に非等価性を報告してこの方法を終了するステップ と、選択されたパス内のすべての強逆結成分について以 て以上のステップを繰り返すステップとを実行すること 【0083】本発明のもう1つの特徴によれば、コンピ プログラム製品が提供される。 前紀スケジュール及び前 郎ピヘイピアは、実行の巡回スレッドを有する可能性が が、スケジュールをスケジュール状態遷移グラフとして グラフとして投現するステップと、前配スケジュール状 修選移グラフ内の強連結成分を識別するステップと、各 **以されていないパスを選択するステップと、選択された** さい場合に、シンポリックシミュレーションを再実行す **一ドを含むコンピュータ可能媒体を有するコンピュータ もる。色部コンピュータコードは、植窓コンピュータ** 

【0084】好ましくは、前記コンピュータコードは、

(20)

(本間2001-142937 (P2001-142937A)

があったコータが、ピヘイビア状態環络グラフの始状態を許够パスリストに割り当てるテップと、評容パスリストに割り当てるテップと、評容パスと、ピヘイビア状態を選択するアップと、非容いンボリックシミュレーションを共行して、カケジュール指弦RTL及びピヘイビア構造RTLとを共行して、カケジュール指弦RTL及びピヘイビア構造RTL内の対応する信号を強切するステップと、過数条件とパングナルの協協がせっでない場合に、状態S」の新しいコピーを指容パスに追加するステップと、過数条件とパングナルの発出。 解なパストに前記値加するステップと、3月からS」への各出路 移ごとに前記追加するステップと、8月からS」への各出路 移ごとに前記値加するステップと、8月からS」への各出路 移ごとに前記値加するステップと、8月からS」への各出路 移ごとに前記値加するステップと、8月からS」への各出路 からンスとなるまで、すべての訪れていない状態について繰り返すステップとを用いて、関わされないが様にのコン

のカットの間のサブ回路と、第2と第3のカットの間の と、等価関係セットの最終集合を、不変項の所望の集合 RTL回路内の対応するカットを職別して、第1と第2 る等価関係を破棄し、1の以上のループ実行について2 つのRTL回路を展開して、繰り返すステップと、前記 関係が同一でなく、かつ、最後のカットにおける等価関 と、前配関係が同一である場合、最後のカットにおける と、鈴価関係セットの集合内で、他のエントリのスーパ 前記コンピュータが、各ループごとに、各カットが前記 スケジュール及びどへイビアのRTL回路における対応 トにおける等価関係が、最後の前のカットにおける等価 係が、最後の前のカットにおける等価関係のサブセット でない場合、最後の前のカットにおける等価関係を、等 **阿関係セットの集合に追加し、1つ以上のループ実行に** ついて2つのRTL回路を周開して、繰り返すステップ として指定するステップとを用いて不変項を抽出するこ ループの各実行の境界における変数値を装すような、ス ケジュール内のパスの構造RTL回路内の3個のカット を購別するステップと、ピヘイピアにおけるパスの構造 するカットの各対における変数どうしの間の等価関係を **歳別するステップと、最後のカットと最後の前のカット** との間の等価関係が同一であるかどうかをチェックする ステップと、前配関係が同一でなく、かつ、最後のカッ 関係のサブセットである場合、最後の前のカットにおけ 等価関係を、等価関係セットの集合に追加するステップ サブ回路が同型であることをチェックするステップと、 ーセットであるすべてのエントリを削除するステップ [0085] 好ましくは、前記コンピュータコードは、

[0086] 【発明の実施の形態】 [4. 好ましい実施形態] [4.1 非巡回の場合の非解数関数によるシンボリックンミュレーション] このサブセクション (4.1) では、有限長の非巡回スレッドを右するスケジュールの場合に、ピヘイピア記述に対してスケジュールを検証する際の、非解較シンボリックシミュレーションアルビリズ

るの母ましい実施例について説明する。このサブセクションで提示されるすべての説明では、メケジュール及びアペイとだけおける有談明のは、メケジュール及びアペイビアにおける有限長の非巡回スレッドのみを比較

【0087】 氷のサブセクション(4.2)では、ルー ブを含むスレッドを比較するという一般的な場合を扱う

検証手続きの好ましい実施例について説明する。 [0088] 4.1.1 スケジュールの支現 この身ましい実施例では、スケジュールは、スケジュー ・ル状態薬移グラフ(スケジュールSTG:Schedule Stu to Transition Graph)の形で指定される。スケジュー ルSTGは、虹銀有限状態マシン(EFSM:Extended Finite StateMachine)やビヘイビが有限状態マシン (BFSM:Behavioral Finite State Machine) 契弘 によく以ている。EFSMについて詳細には、K.T. G.

「BFSM: behavioral Finite State Machine) 表現によく似ている。 F SMについて詳細には、 に、 に an and A.S. Krishnakumar, "Automatic functional test generation using the extended finite state monthine model", in Proc. Dorsign Automation Conf., Jun e 1993、を整隅。 BFSM英別について詳細には、 Q. Lakshminarayana, A. Raghunathan, and R. K. Jha, "The corporating speculative execution into scheduling for control-flow intensive behaviors", in Proc. De sign Automation Conf., pp. 108-113. June 1998、を整備。 FSM英別について詳細には、 W. Wolf. A. Takach, C. Huang, and R. Mano, "The Princeton University behavioral synthesis system", in Proc. Design Automation Conf., pp. 182-187, June 1992、を参照、スケジュールSTGは、機能RTLコードが各状態に望め込まれた、状態磁格グラフからなる。

[0089] STGの状態S<sub>1</sub>内に埋め込まれたコードは、S<sub>1</sub>で実行されることが必要なデータバス高算を指定するとともに、S<sub>1</sub>からのそれぞれの川状態道移に対する遺移条件を計算する。状態道移グラフの状態内に則め込まれたコードは、変数V、消算O、及びクロックにより定義することができる。クロックは、変数の値の更新を支配する。

[0090] 変替は、V=(P1, P0, R, T)のように4つのセットに分けられる。P1はプライマリ入りのセットであり、P0はプライマリ出かセットであり、Tはレジスタ変数のセットであり、Tは一時変数のサットである。前算は、間間高質のセット、及び、代入資率のセットを含む。間関は第の程行を間倒するために使用される。代入前算は、変数に置き代入し、あるいは、災数の値を変更する。各領算のpiは、それに関連づけられた対応する条件に言を有する。条件には隔壁式であり、他の代入及び関層高質の結果を用いて偶成されることが可能である。減算のpiは、eが其の場合に限り送行される。状態固移は、状態内に埋め込まれたコード内のgoto文によって指示されることも可能である。それぞ

Lのこのようなgoto文の実行条件は、対応する状態選移

【0091】スケジュールは、明確に定義された始状態 は、始状態で開始し、終状態で終了する。なお、複数の 可能な終状態がある場合は、スケジュールのすべての終 ルに追加し、このダミー状態のみを終状態と見なすこと **火値から入ってくる弧を有するダミー状態をヌケジュー** 及び終状態を有すると仮定する。スケジュールの実行 によって、単純な場合に帰着される。

さまざまなハイレベル合成ツールが、関御フローグラフ ラフ (CDFG)、及びピヘイピア有限状態マシン (B FSM)を含むピヘイピア記述に対するさまざまな表現 を使用している。本発明は、スケジュール表現の特定の 方法に制限されない。これら及びその他のうちのいずれ の表現も、検証手続きのために使用可能である。説明の ビア及びスケジュールを表すために同じデータ構造を使 (CFG:control flow graph)、データフローグラフ 何単化及び一質性のために、好ましい実施例は、ピヘイ (D.F.G:data flow graph) 、制御ノデータフローグ . 2 ビヘイビアの表現 [0092] 4...

ケジュールに対する一貫したデータ構造を有することに [0093] このため、好ましい実施例では、ピヘイビ して表現される。どへイビアSTGは、どへイビアから 直接に導出されることを除いては、上配のスケジュール STGと類似している。ループを含まないビヘイピアの 部分は、単一の状態にまとめられる。従って、非巡回ビ ヘイピアは、ただ1つの状態を有するピヘイピアSTG に開訳することができる。このようにピヘイビア及びス より、等価性チェックの問題は、スケジュールSTGと **アはビヘイビア状態遺移グラフ(ビヘイビアSTG)と** [0094] · K~4 K7 STG (BSTG) & UX 7 ピヘイピアSTGの等価性を証明することに帰着する。 Va-NSTG (SSTG).

・BSTGとSSTGにおけるプライマリ入力変数とう しの間の対応。

が与えられた場合、目標は、対応する出力変数において BSTG及びSSTGによって生成される値が等しいこ とを証明することである。

レッドどうしの間の等価性を証明するという惻假された 問題にある。SSTG (あるいはBSTG) における実 行のスレッドとは、角状値に始まり終状値に終わる状態 は、状態遷移グラフ内のサイクルを一定有限回通ること 焦点は、BSTG及びSSTGにおける実行の関々のス パスが単純であることは要求されない。 すなわち、パス [0095] 前述のように、この好ましい実施例では、 **煙移グラフにおける有限長のパスのことである。なお、** り可能である。

[0096] SSTG (あるいはBSTG) における実

は、構造グラフに変換される。

荷算を実行するハードウェアコンポーネントを表し、辺 有向グラフG= (V, A) であって、頂点のセットVは のセットはコンポーネントの構造連結性を要すものであ る。 所点∈ Vは、 型属性を有し、これは以下の値をとり [0097] 定義1 (構造グラフ) : 構造グラフとは、

・IN (プライマリ入力変数と、レジスタ変数の現サイ クル値とを教す)

・OUT(プライマリ出力変数と、レジスタ変数の次サ

イクル値とを表す)

[0102] BDDは、条件付き等価関係に関連する条

 $V \cong \{(u_1,c_1),\ldots,(u_n,c_a)\}$ 

作を表すために使用される。一般に、条件自体は、入力 変数で表すことも可能であり、また、さまざまな算格及 び条件資算の結果を含むことも可能である。しかし、条 作は、INノードに加えて、OP及びMUXノードの出 される。実際、BDDは、制御論型に対してのみ構成さ れる。これは、POに送られる次状館論理Rstate-noxt と、MUXノードを涵るどのパスがセンシタイズされて

・OP(算術演算及び比較演算を含む、アトミックなワ ードレベル(資料を表す)

・LOGIC (制御またはランダム輪理を表す)

構造グラフ内の辺にはそのピット幅が標配(annotate) さ ·MUX

【0098】計算のセットから構造グラフを構成するプ

いろかあるいはマルチファンクションFUがどのように

ムの好ましい実施例の撥似コードを図4に示す。アルゴ

の筍値関係からはじまる。このアルゴリズムは、POノ リズムは、SSGFLBSGFのINノードどうしの関 ードに到達するまでSSGtMの中間信号を通って条件 けき等価関係を生成し伝数させ、SSGT及びBSGF

[0103] BSGr とSSGrを比較するアルゴリズ

股定されているかを決定する胎理とを含む。

カで (これらをまとめて、「基底変散で」という) 表現

ロセスは、ハードウェア配述目語(HDL:llardware D イマリ入出力変数、定数値、及び、レジスタ変数の現サ イクル及び次サイクルの値を表すように生成される。O て生成される。単一ピットまたはピットベクタに対する される。これらの条件に対応するOPまたはLOGIC 実行される代入を決定するために、MUXノードへの選 ることと類似している。1N及びOUTノードは、プラ 比較商類、case商算など)に関連する代入前算に対応し が、相異なる条件下で同じ変数に代入を行うときに構成 ノードの出力は、与えられたクロックサイクルにおいて ブール演算の使用により、構造グラフ内のLOG 1 Cノ escription Language) からハードウェア構造を推論す Pノードは、ワードレベル計算及び条件資算(例えば、 ードが生成される。MUXノードは、相異なる代入文 択(セレクト)入力として使用される。

**単格をとることによって、行われる。** 

[0104] まず、順序セットArr1 (Arr2) を、BSG

における出力信号どうしの関の無条件等価件をチェック

る。後方程さ優先探索走査を用いて、各ノードは、その 禁移ファンイン(transitive fanin)内のすべてのノード の後にのみ現れるようにされる。水に、BSGF内の基 庇変数を、PI、OP、及びMUXノードの川力として

「 (SSGr) 内のすべてのノードを含むように構成す

[0099] SSTG内の実行のスレッドTと、韓価で する実行のスレッドT'とが与えられると、各スレッド あることを証明することが要求されるBSTG内の対応 る。こうして、問題は、2つの構造グラフSSGTとB に沿って曳行される計算はまず構造グラフに変換され SGr の等価性を証明することに帰着する。

[0100] このセクションの残りの部分では、以下の **性質を利用した、構造グラフの等価性チェックのための** ・どへイビア配述からスケジュールを生成するときにO アルゴリズムの好ましい実施例について説明する。

の条件付き等価関係を表す等価性リストに関連づけられ

SGfノードは、その川力と、BSGr.内の信号との間 る。等価性リスト内のエントリは対(u, c)である。 ただし、uはBSGr信号の識別子であり、cは、等価 性のための条件を表すRDDである。BSGr'とSSG Tの入力どうしの頭の対応を用いて、SSG1内のINノ **査し、各ノードを、その入力から出力へ等価性リストを** 伝散させるように処理する。OP、LOGIC、及びM

> ・算術変換(例えば、分配則や、乗算をシフトと加算で 置き換えることなど) は実行されないということ。 Pノードのアトミック性は保存される。

開号vがBSGT'内の俑号u1, u2, ..., unに条件付 げのスレッドが与えられると、その中で実行される計算 ∞ n (条件とは、BSG+あるいはSSG+内の入力変数へ [0101] 定義2 (条件付き等価性) : SSGp内の きや価であるとは、対応する条件で1, c2, ..., c

(22)

アルゴリズムは、BSGF内の対応するOUTノードで 等価性が確定しているか、及び、対応する条件がトート 合、アルゴリズムは、SSGTとBSGr は勢値でない と報告する。SSGrのすべてのOUTノードに対して 熊条件等値性が得られた場合に限り、アルゴリズムは、 ロジーであるかどうかをチェックする。そうでない場 SSGfとBSGrが労働であると質問する。

の出力を要すようにBDDを構成する。これを行う理由 [0105] 等面関係は、OPノードを通って以下のよ を実行するBSGrrMのOPノードuで、vの入力がu の対応する入力と条件付き等価関係を有するようなもの が存在する。このような場合、vとuの出力は、対応す **一ドに澄渇した場合、等価性リストをその川力に伝版さ** せるのではなく、BSGr.内の場底炎散の関数としてそ は、LOGICノードはスケジュールにおいて変換また 価性を証明するためには解釈される必要があるからであ る。2 人力MUXノードの1 (0) データ人力からその 川力へ等面性リストを伝搬させることは、遊状 (セレク と、データ人力の等価性リスト内のすべての条件との論 ろ入力の等価条件の輪型損と等値である。 LOG1Cノ うに伝数する。SSGT内のOPノードvと、同じ函数 は導入されることがあるため、SSGrとBSGr の等 ト) 信号に対するBDDを取得し、それ(その値数)

**【0106】 [4.2 …散的な場合のスケジュール検 低アルゴリズム| このサブセクションでは、一般的な場** ピアの川力間の無条件等面性を確定することである。S TGが非巡回(無関路)であれば、セクション4.1の ツンボリッケツベコフーションに残んへや宿在チェッゼ で十分である。フィードバック(ループ)が存在する場 は、ループが完了するまで反復せずに2つの記述の等価 性を検証することが必要である。ループを扱うため、プ ルゴリズムは、ルーブ不変項を加出する。不変項は、ル ープ終了点におけるスケジュールとどへイピアの間の変 数対応である。不変項補川は、等価性の証明を生成する ためにループを完了まで反復することを不要にする自動 帰謝法に基づく。すべてのループ不変項が抽出されない 場合、等価性チェッカはフォールスネガティブを返す可 ーリングが前に定義した意味で代表的である場合、すべ ことがないという意味で安全である。本発明の検証アル ゴリズムについて説明するための何を提示し、その後で このアルゴリズムのタスクは、スケジュール及びピヘイ 胎性が高い。 等価性チェックアルゴリズムは、スケジュ てのループ不変項を検出し、英の否定及び肯定を返すこ とを保証する。このアルゴリズムは、関った肯定を返す **合のアルゴリズムの好ましい実施例について説明する。 育、等価性チェックアルゴリズムが有用であるために** 

> 識別する。次に、Arr1を通る走査を実行し、場底変数に 対応しない出力を有する各ノード (すなわち、各LOG ICノード) について、そのノードの出力に対するBD Dを、その入力におけるRDDに関して取得する。各S

アルゴリズムは、入力として、ピヘイピア及びスケジュ [0107] 4.2.1 具体例

その詳細について脱男する。

一ドに対する等価性リストを生成する。次に、Arr2を走

UXノードを通って等価性リストを伝数させる技術につ

いては後述する。SSG<sub>T</sub>のPOノードに到達すると、

の値代入の空でないセットをを表す)であって、条件c

kの下で、SSGTMの信号vにおける質が、BSGTM 条件c1, c2, ..., cヵが存在する場合をいう。条件付

の信号ukにおける値と等しいことが保証されるような

き等価関係を表すために、配法

[数1]

(寺間 2001-142937 (P2001-142937A)

して吸供される。アルゴリズムは、SSTG内の小さい パスセットを列挙することによって動作する。これらの ールの状態圏移グラフ (STG) 装現 (それぞれBST G及びSSTG) をとる。STGに加えて、プライマリ 入力及び川力の対応のリストも、アルゴリズムに入力と パスは、SSTGとBSTGの間の等価性を証明するた めの基礎として使用される。

の例を示す。このSSTGに対して、次の状態列を列萃 [0108] 例3:図5 (a) に、函約(reduced) S S TG (強連結成分を抽出し無閉路パスをつぶしたもの) することができる。

(AE, ABCE, ABCDCE, ABCDCDC

のパスのすべてをBSTG上でシミュレートする必要は は、ループ本体の異なる回数の実行に対応する。これら なお、パス [ABCDCE, ABCDCDCE, ...] ない。ノード(C)は、ルーブ粒子点に対応する。

これらは、ループに全く遭遇しないこと、ループ終了条 に対する問題を設定することである。従って、この例で プ本体を 2 回実行することによってループ不変項の生成 る。終了パスが3回現れるパスを列挙する理由は、ルー 作に遭遇するがループ本体には遭遇しないこと、及び、 ルーブ本体を2回実行すること、にそれぞれ対応する。 (C)がの、1、及び3回現れるパスの数を列萃する。 最初の2つは単純パスであり、明確に列挙すべきであ 列挙されるパスは (AE, ABCE, ABCDCDC

より取得する。次に、アルゴリズムは、SSTG及びB DC) 及び (ABCDCDC) における対応のセットは [0109] これらのパスのそれぞれについて、BST G内の対応するパスをシンボリックシミュレーションに STGの対応するパスが等価であることを証明する。ル ルゴリズムはさらに遡み、ループ本体内の徴算が任意回 数実行された場合に、SSTG及びBSTGの対応する パスどうしの間の等価関係が依然として維持されるかど うかを帰納的に証明する。これを行うため、アルゴリズ での変数対応を抽出する。この場合、カット点(ABC ープ本体を含むパス [ABCDCDCE] に対して、ア を任意関数だけ(ABCDCDC)に連接しても依然と て、 (ABCDCDCE) と、対応するBSTGパスが ムは、カット点 (ABCDC) 及び (ABCDCDC) 同一のままである。従って、帰納法により、列(DC) して変数対応は維持されるということができる。従っ 等価である場合、ループ本体の任意回の反復に対して

るために必要十分である。

[0110] 次に、ルーブ棒子に対応するカット点での 別のシナリオを考える。セットが変わると、観った結果 対応する変数のセットが同一のままにならないような、 を避けるために、収攻するまで反復する必要がある。 も、SSTGとBSTGは依然として等価になる。

てこれらのいずれを用いても、bとgが等価であると見 は、それぞれ、回路のどへイビア及びスケジュールを示 す。なお、これらの2つは、スケジュールの状態5にお ける文c=d+2のため、対応しない。最初に離別され 3, 4, 5, 2, 6) である。図6(b)における状態 2は、ループ終了として臨別され、状態3、4及び5は なされていることによりフォールスポジティブを引き起 ループ本体として識別される。 ループの1回の実行のシ ンポリックシミュレーションの後、得られる変数対応は {a ≒ p, b ≅ q, d ≅ s | であり、2回の実行の後に イブ (臥った肯定) を避けるために収束するまで反復す るループ本体を含むパスは (1, 2, 3, 4, 5, 2, (out = b) をシミュレートするための対応セットとし る必要がある理由を何示する。図6の(m)及び(b) は〔b≡q〕である。なお、ループの後の状態6の文 こすことになる。

非等価であると見なすことができる。このように、この qが対応しないと判定し、スケジュールとどへイビアは 場合、収束に到達するには、最初のパスのもう1回の反 仮をシミュレートしなければならない。 すなわち、パス 【0112】3回の実行の後にはじめて、手続きはbと [1, 2, 3, 4, 5, 2, 3, 4, 5, 2, 3, 4, 5, 2, 6) もシミュレートする必要がある。 【0113】例5:最後にもう1つのシナリオを考える は、収束に到達するまで単調に減少した。しかし、一般 対応セットに対してシミュレートされるのに加えて、す ペイのこのような樹小セットに対してシミュレートされ に、反復を多く実行するにつれてこのセットが任意に変 クすることである。従って、収束が得られるまですべて は、他の対応セットのスーパーセットでないセットに対 **広する。ループ本体に続くコードは、収束後に得られる** る。後述するように、これらの極小対応セットを用いた シンボリックシミュレーションは、等価性をチェックす ループ本体のすべての反復回数について特価性をチェッ る。なお、この追加の対応によりシミュレートしないこ 化するような例が考えられる。なお、実質的な目標は、 とによってフォールスポジティブを生じる可能性があ 必要がある。前の例で、ループ反復後の対応のセット の極小対応セットを追除しなければならない。これら

ことである。ループ不変項をループ終了点で計算する必 て見川される。各SCCは、1個以上の終了ノードを有 図7は、一般的な場合を扱う本発明の方法の好ましい実 脳例の擬似コードを示す。このアルゴリズムの第1のタ スクは、SSTGの、ループを構成する部分を識別する strongly connected component) を識別することによっ **₩があるからである。ループは、強連結成分(SCC:** し、そこからSCCの外へ選移することが可能である。 [0114] 4, 2, 2 アルゴリズムの詳細

その後、SCCを通らないサブパスを併合して、以後列

5,

[0111] 例4:図6に示す例は、フォールスポジテ

得問2001-142937 (P2001-142937A)

よってSSTGがどのように影響されるかが示されてい る。これらのステップの結果、状態には、状態に及びD **挙する必要のあるノード及びパスの総数を減らすために** SSTGをつぶす。図5 (a) に、これらのステップに からなるSCCの終了点として疑別される。

【0115】図7における擬似コードの第4行は、簡約 る。このパスは、BSTG内の対応するパスに対してチ ェックされなければならない。パス列挙前にSSTGを (a) のSSTGの場合、最初に列挙される3個のパス 節約することにより、大幅にパスは少なくなる。図5 SSTG内のパスを列挙するループの開始をマークす

とは、本明細番においては、状態遷移によりある状態か これらのパスのそれぞれについて、BSTG内の対応す る。図7の疑似コードの第5行は、列挙されたSSTG 第5行は、また、SSTGパス内の核了ノードへの各避 図7の疑似コードの第6行は、SSTGパスを列萃する のに必要なすべての状態選移判定をカプセル化するパス シグナル(Pathsignal)という信号を生成する。SSTG パスに対応するBSTG内のパスを確別するシンボリッ 移に対応するRTL回路内のカットを織別する。カット パスに対するRTL回路(SSGという)を取得する。 ら別の状態へ伝볞する変数のセットとして定義される。 クシミュレーションは、図1の疑似コードの第1行で、 手続きConstrained#symbolic#simulation()によって、 るパスをシンボリックシミュレーションにより取得す (AE, ABCE, ABCDCDCE)

がある。

のルート状態からはじめて、そのタスクは、Pathsignal 8の第5行)。 水に、その状態からの、Pathsignalと両 立する出遷移を職別する。このプロセスは、BSTG内 【0116】図8を参照しながら、Constrained#symbol と両立する遷移により到達可能な状態を職別することで ある。到違した各状値で、対応する信号を職別するため に非解釈シンポリックシミュレーションを実行する (図 ic#simulation()の詳細について説明する。BSTG内 のEND状態に到達するまで鋭く。

は、得られた出力等価性が、パス条件以外の条件付きで テップは、列挙されたパス内のループから不変項を抽出 することである (図7の第8~12行)。 このステップ は、ループが存在しないときには不要となる。パスに沿 実行しなければならない。図7の第13行及び第14行 って遭遇する各SCCに対して、図9に記載した手続き ストのうち、前のシンボリックシミュレーションの結果 としてSCCカットで得られた変数対応より小さい各対 応セットに対して、図7の第12行に示すように、SC Cに続くパス部分のシンボリックシミュレーションを再 [0117] 図7の全体アルゴリズムに戻って、次のス 変数対応セットをcorrespliset出listとして返す。このリ return#loop#invariants()を呼び出す。この手続きは、

場合、STGは等価でないと見なされる。川力が、列塔 あるかどうかをテストする。そのように条件付きである されたすべてのパスに対して無条件に等価である場合、 STGは等値であると見なされる。

[0118] 図9を参照すると、return#loop#invarian ts()への入力は、列挙されたパスにおいてSCCの終了 ットのリストを返す。図9の手続きの第1行は、ssyftcu /一下に撤退する3つのインスタンスに対応する55G とカット 3 の間の回路が、カット 1 とカット 2 の間の同 の手観きは、ループに続くパス部分のシンボリックシミ ュレーションが実行されなければならないような対応セ 取得する。第2行及び第3行は、BSGにおいて専川さ れたカットどうしの間の2つの回路を取得し、カット2 路の単なる別のインスタンス(コピー)であるかどうか 適当なcorresp#sot#listが返される。カット間の恒路が 内の3個のカット (ssg#cuts 1, 2, and 3) である。こ tに対応するBSGにおける変数 (hsghoutsという) を 同型である場合、非自明な対応セットが存在する可能性 を確かめる。そうでない場合、対応がないと見なされ、

は、次の状御列からなる。

対応セットをcorresphsotHistに追加する。これは、例 [0119] このセットを見つけるため、手殻をは、最 は、この点で見出した対応リストをcorrespilsot#listの 一部として返す。そうでない場合、corresp#setn:1を切 レーションを繰り返す。この手続きはまた、1回の実行 のシンボリックシミュワーションにおける対応セットの 使用により新たな変数対応が生成されたときには、この 4及び例5において厳酷したようなフォールスポジティ p#sotn+1)を、そのシミュレーションの最初における対 開変数対応として、1ループ実行のシンボリックシミコ 各シミュレーションの最後に得られる変数対応 (corres 応 (corresp#set<sub>n</sub>) と比較する。これらのセットが同一 に、SSGとBSGをシンボリックシミュレートする。 切にカット2からはじめて、一度に1回のループ実行 である場合、これは要求された固定点であり、手続き (すなわち、2つのカットの間の部分) だけ溢むよう ブを避けるためである。

SSTGパスのPathsignalを用いて実行される。

[0120] すべての変数対応を織別するのに要する反 最悪の場合、これは、SSGとBSG内のループ本体の 変数の個数に関して線形であり、ほとんどの対応は、最 は、有限回の(実際には、非常に少ない)ルーブ反復で 変数の個数の相になりうる。実際には、変数対応の数は 復回数は、可能な変数対応の総数によって開限される。 初の実行自体の後に見出される。従って、この手続き ループ不変項を得る手段である。

[0121] 4. 2. 3 アルゴリズムの正当性及び有

フォールスポジティブは、2つの接現が実際には等価で ないときに、検証ツールがそれらを等価であるとみなす 場合に生じる。フォールスネガティブは、2つの表現が

2

(54)

実際には等値であるときに、検証ツールがそれらを等値 でないとみなすときに生じる。次の定理は、本発明のア ルゴリズムを特徴づける。

(\*) 「代表的」スケジューリング、及び、(b)実現 不可能な反復カウントによるネガティブの可能性がな [0122] 定理2:図7の手続きCompare#STGsは、

ずれも非巡回的であるとき、フォールスポジティブが発 [0123] (証明) スケジュール及びビヘイピアがい 生しないことは、基本的なシンポリックシミュレーショ い、という仮定の下で、フォールスポジティブまたはフ ンに基乙へ等価性チェッカの性質である。フォールスネ ガティブは、非巡回的である場合、シンボリックシミュ レータによって非解釈とされる演算の機能の知識が最適 **は、ツンボリックツミコフーションに揺んへ特度右チェ** ッカは、非巡回パスにおいて正しい変数対応を見出すと 化で使用されるときにのみ発生しうる。残りの解析で オールスネガティブを発生しないことが保証される。 いう作実に依拠することができる。

場合に本発明のアルゴリズムでいつフォールスネガディ [0124] さらに興味深いことは、ルーブが存在する **ル記述は非巡回的 (ループを含まない) である場合、あ** るいはその逆の場合は、代表的スケジューリングによっ ブ及びポジティブが発生し得るかの解析である。ピヘイ ピア記述は巡回的(ループを含む)であるがスケジュー て許容されない。両方の記述にループがある場合、生成 される変数対応が多すぎるときにフォールスポジティブ が起こり、生成される変数対応が少なすぎるときにフォ ールスネガティブが起こる。

ブ停止の正当性は、本発明の手続きでは、スケジュール 対応を確定することによってチェックされる。本発明の 可能性がある。また、実現不可能な回数の反復の後にの であり、かつ、実現可能な反復カウントがないという仮 [0125] まずフォールスネガティブを考える。ルー **及びどへイビアにおけるループの停止条件どうしの間の** アプローチは、実現不可能な反復カウントについて知ら ない。 スケジュール記述を生成するために使用される最 **合、本発明の手続きはフォールスネガティブを報告する みループ本体どうしの間の差が「活性化」されるときに** もこれは起こり得る。従って、代表的スケジューリング 定の下では、このようなフォールスネガティブは起こり 商化が実現不可能な反復カウントの知識を使用する場 得ない (証明故)。

で、ループを収束するまで反復する。このプロセスで生 る。ループのn回の実行後に得られる変数対応セットを 反復のたびに変化しない変数対応のセットを識別するま 成される各種小変数対応セット(これは、他の対応セッ トのサブセットではない)は、ルーブ終了点の後のコー ドのシミュレーションを実行するために別々に使用され [0126] 図9に示すように、不変項加出手続きは、 CSnで表す。以後のシミュレーションで使用される、

プ終了以後のシンボリックシミュレーションのために使 ち、CSnG (CSi) は、極小対応セットであり、ルー 桶小変数対応セットの集合を (CSi) で要す。すなわ

[0127] 明らかに、CSn⊆ (CSi) でのシミュレ **ーションにより生じるネガティブの等価性結果は、実現** 不可能な反復カウントがないという仮定と、シンポリッ クシミュレーション手続きの基本的性質により、真のネ ガティブである。 【0128】次に、フォールスポジティブを考える。変 対応のセットであるということができる。従って、CS る。帰納法により、このセットは、n M k に対して、n 回の反復に対応する実行されたパスに対する正しい変数 価性結果は、n≧kのすべてのnに対して真のポジティ kでのシミュレーションの後に得られるポジティブの等 数対応のセットの固定点(すなわち、CSk=CSk+l) に到達するのに k + 1 回の反復が必要であると仮定す ブである。 【0129】ここで、スケジュールにおけるkより少な 20 い同数のループの実行に対応するパスを考える。すなわ ち、nくkとする。アルゴリズムは、n回の反復後の終 ち、この場合にフォールスポジティブがないことを示す 丁をチェックするように正しく動作すること、すなわ 必要がある。考慮すべき衣の2つの場合がある。

合、すべての極小対応セットは、ループ終了点以後明示 [0130] 1. CS<sub>n</sub>は、桶小対応セットのうちの 的にシミュレートされるため、CSnはフォールスポジ 1つである。すなわち、CSnGCSiである。この場 ティブを発生し得ない。

のいずれかの対応セットでのシンボリックシミュレーシ [0131] 2. CSnは、極小対応セットのうちの **たの丝塔 カットかのツンボリックツバコフーツョンだお** ュレーションをする必要はない。 (他方、 [CS<sup>i</sup>] 内 (CSi) 内の対応セットのうちの1つのスーパーセッ トでなければならない。この場合、 (CSi) 内のすべ ジティブの結果を発生する場合、CSnでのシンポリッ り、CSnでのシンボリックシミュレーションは意味が クシミュレーションも間接となり、CSnで別属のシミ 1つでない。極小対応セットの定義により、CSnは、 ョンがネガティブを発生する場合、配近は非等価であ

入るたびに不変項を解析しなければならない。 前述のCo mpare#STGs手続きに加えて、ループネスティングを決定 する解析が必要となる。ネストしたループをどのように 扱うかについての直観的な説明は、セクション5.2の [0132] [4, 2, 3, 1 ネストレたループの限 いしネストしたループを扱うためには、内側のループに ネストしたループの例のケーススタディを参照。

ズムの効率は、基本的に、次の3つのファクタから導き 【0133】 [4.3 アルゴリズムの効率] アルゴリ

(56)

特 [3]2001-142937 (P2001-142937A)

出される。

- (1) データパス状態は列挙されない。
  - (2) 算術は解釈されない。
- (3) 不変項を抽出するためにループは完了まで反復さ

税きCompare#STGs()におけるSSTG内のSCC額別及 び終了点の観別は、SSTGのサイズに関して根形であ 関して2次になり得る。実際には、スケジューリングに 【0134】ファクタ(1)及び(2)は、等価性チェ ックのためのアルゴリズムの内側ループで使用される非 (3) は、本発明の不変項抽出アルゴリズムによる。手 る。Compare#STGs()におけるパス列挙は、つぶしたSS FGに対して行われる。これは、最悪の場合に列挙され **ール内のループの個数に関して指数関数的になることを** 意味する。このパス数は一般に非常に小さい可能性が高 い。不変項を加出するために、列挙に要するループ反復 5。 最悪の場合、これは、ループ本体内の変数の個数に るパスの個数が、SSTG内の状態数ではなくスケジュ 解釈シンボリックシミュレータに含まれる。ファクタ 回数は、可能な変数対応関係の個数によって関限され

性チェッカ15.7は、前記スケジュール構造グラフと

前配ピヘイピア構造グラフの等価性をチェックする。

取り和んでいる特定の検証問題に対する高速なカスタマ すべての変数対応は、2回のルーブ反復の列挙により見 が、このようなサブ回路は実際は非常に小さいため、B DD生成がボトルネックとなることはない。 シンボリッ 技術に比べて、本発明のアルゴリズムのランタイム計算 算には二分次定ダイヤグラム (BDD) が必要とされる **垂される。 シンボリックシミュレーション中のブール街** おけるエラーがないときの代表的スケジュールの場合、 クモデルチェック (symbolic model checking)のような 最 (複雑さ) は小さいため、木苑明のアルゴリズムは、 イズされた解法として適している。

等価性証明器14.3は、非巡回スレッドの等価性を証 イビア配述から得られるような回路のスケジューリング のようなシステムの好ましい実施例を図15に示す。ル **一ブ不変項抽川器14.1は、ルーブが存在するときに** 非巡回スレッドの十分なセットを決定する。シンポリッ ム] 本発明の重要な特徴は、回路のスケジュールがピヘ クシミュレータ14.2は、ループ不変項を加出する。 [0135] [4.4 スケジューリング検証システ の正当性をチェックするシステムとして実現される。

明する。このシステムは、 ・サイクル境界の導入

・預算並べ替え

・ループの原間、巻付け、近畳み及びパイプライン化 ・預算の投機実行 のうちの1つ以上により変換されたピヘイピア配述を扱 うことが可能である。

システムとして実現される。このようなシステムの好ま [0136] 本発明のもう1つの重要な特徴は、回路の アヘイピア記述に対して回路のスケジュールを検証する

しい実施例を図16に示す。スケジュール状態選移グラ **フ生成器15.2は、15.1からスケジュールを受け** 取り、スケジュールをスケジュール状態盛移グラフとし 取り、ピヘイピア状態器移グラフから対応するピヘイピ ピアスレッドをピヘイピア構造グラフに変換する。等価 は、回路のピヘイピアをピヘイピア状態遷移グランとし は、スケジュールを受け取り、スケジュール状態圏移グ ラフから実行のスケジュールスレッドを題択する。どへ イピアスレッドセレクタ15.5は、ピヘイピアを受け アスレッドを踏択する。変換器15.6は、スケジュー て指定する。ピヘイピア状態選移グラフ生成器15. 3 ルスレッドをスケジュール構造グラフに、及び、ピヘイ て指定する。スケジュールスレッドセレクタ15.4

路のスケジューリングの正当性をチェックすることを可 内の命令は、ルーブが存在するときに非巡回スレッドの **一タシステム] コンピュータは、本発明の技術を実現す** るための非常に有効な手段である。木発明の技術を実現 するこのようなコンピュータシステムもまた木発明の技 析的範囲内に入る。このようなコンピュータは、プロセ ッサ及びメモリを打する。メモリは、コンピュータが回 十分なセットを決定するためにループ不変項を抽出する 命令を含む。さらに、命令は、ルーブ不変項を抽川する らに、命令は、非巡回スレッドの等価性を証明する命令 【0137】 [4.5 スケジューリング位証コンピュ そのアヘイアア智滋から仰られる。 具体的には、メモリ ためのシンボリックシミュレーションの命令を合む。さ 能にする命令を含む。ここで、同路のスケジュールは、 を含む。

[0138] なお、コンピュータは、PC、メインフレ モートコンピュータを含むいかなる種類のコンピュータ **ーム、ワークステーションあるいはネットワーク上のリ** とすることも可能である。

この命令は、コンピュータが、図4に示した擬似コード コンピュータが、図1~図9に示した擬似コードを、単 独に、またはすべての可能な組合せで、実行することを 可能にする命令を含むメモリを有するコンピュータから 【0139】コンピュータシステムの好ましい実施例 を実行することを可能にする。別の好ましい実施例は、 は、命令を含むメモリを有するコンピュータからなる。

アセンブリ吾語及び機械語を含む(これらに限定されな [0140] なお、命令は、高水神智語、低水神智語 い)任益の形式とすることが可能である。

製品は、コンピュータが回路のスケジューリングの正当 性をチェックすることを可能にする命令を有するコンピ **ータプログラム製品として実現される。このプログラム** 【0141】 [4.6 スケジューリング検証コンピュ **ータプログラム製品]本発明の電要な特徴は、コンピュ** 

ューケゴ酸媒体を含む。なお、コンピューケゴ酸媒体は、フロッピー(単縁高弱)ディスク、ハードディスク、CD、チップ、テープ、1 C(付きカートリッジなどを含む(これらに吸定されない) 行題の固定媒体を含む。コンピューケ可凝媒体は、ネットワークを通じて伝送される、あるいは、インターネットからグウンロードされる命令も含む。

[0142] 好ましい実施例では、コンピュータコードは、コンピュータが回路のスケジューリングの正当権をチェックすることを可能にする。ここで、回路に対するスケジュールは、カインコールは、ループが存在するときに非巡回スレッドの1分なセットを次定するためにループ不変項を抽出するためのコンピュータコードと、オ巡回スレッドの等価性を指明するコンピュータコードと、非巡回スレッドの等価性を指明するコンピュータコードと、キージューケー

[0143]コンピュータプログラム製品の好ましい状態例は、コンピュータコードを合むコンピュータ可放戦体を合む、このコンピュータコードは、コンピュータが、図4に示した機似コードを光行することを可能にする。別の好ましい実施例は、コンピュータが、図7ー図9に示した機似コードを、単独に、またはすべての可能な紹合せて、実行することを可能にするコードを含むコンピュータが可能体を含むコンピュータブログラム製品からかる。

[0144]なお、コンピュータコードは、資水停ぎ路、低水降音器、低水降音器、アセンブリ質語及び機械語を含む(これらに限定されない)任意の形式とすることが可能であ

[0145]

「孫明の効果】 | 4. 7 結果:ケーススタディ| 本発 明を適用した結果について、本発明のアルゴリズムを投 みの実際のスケジューリングの倒に適用した詳細なケー ススタディの形で槌示する。これらの設計は、状値変数 の個数及び算術商算の計算量(複雑さ)に関して十分に 大きいため、状態マシン等価性やシンボリックモデルチ エックに基づく往来のBDDによる検証アプローチでは 確実に失敗する。

[0146] 4...7..1. 数機之化ジューリングの的 因10 (a) に示すピヘイピアSTGを考える。このピヘイピアを、投機支付及びルーブ変換を合む吸筋のスケジューラによってスケジューリングした。結果として得られたスケジュールングした。結果として得られたスケジュールSTGを図10 (b) に示す。ピペイピア州のルーブを考える。定常状態では(すなわち、ルーブが多数回光行されると仮定すると)、スケジュールSTGは、状態S6からなるSCCをたどることに注信する。スケジュールの解析を実行することに注して、流光状態では、ルーブの類たな質がクロックサイクル

ごとに開始され、大きなパフォーマンス改善につながる

ことを示すことが可能である。

の状態ビットを含む。R. K. Brayton et al., "VIS: A s IID L 記述は、289行のコードからなり、47個の資 ystem for verification and synthesis", in Proc. In 及び8頃の中間変数(プライマリ入力及びプライマリ出 とスケジュールの構造同型チェックでは、これらの幹価 性を証明することはできない。 (データパス+制御) 状 ば、ゲートレベル)のF SM等価性チェックツールでは 非常に困難になる。 ピヘイピアSTGは250個の状態 この例は、セクション1.2で首及したほとんどの最適 ントの複製、ループパイプライン化、及び投機実行)を ルの複雑さを大幅に増大させる。 ピヘイピアSTGのV ||DL記述は、122行のコードからなり、7個の演算 力を除く)を含むのに対して、スケジュールSTGのV 化(サイクル境界の導入、商類の並べ替え、パスセグメ 同時に含むことである。これらの最適化は、スケジュー 算及び54個の中間変数を含む。 明らかに、ピヘイピア ピットを含み、スケジュールSTGはさらに大幅に多く [0147] 検証の観点から、この例で興味深い点は、 態空間をたどるVSIのような従来の低レベル (例え t. Conf. Computer-Aided Verification, July 1996,

[0149]シンボリックシミュレーションがどのように進行してループ不変項を検出するかを輝くるために、ここでは、スレッドT1=(SA, SB, SC, SD, SE, SF, SF, SG)を考える。スケジュール構造グラフSSGTを図1に示す。スケジュール内の状態境界に対応する構造グラフMのカットは高線を用いて示される。

[0150] 手級きConstrained#Symbolic#Simulation は、ピヘイピア、及び対応する構造グラフにおける、対 応するスレッドを自動的に加出する。結果として得られ るピヘイピアスレッドは「2= (SO, S1, ..., S 7, S1, ..., S7, S1, S8) であり、その構造 グラフ(BSG<sub>12</sub>)は図12に示される。なお、スケジ

(28)

特間2001-142937 (52001-142937人)

ば、SSGTI内の第1のカットにおける信号に関する等 **何思係は、(14, 14, 1), (M2, M2, 1), (c, c, 1), (i#** y (s1, s2, cond) は、SSGT1内の信号s1とB 意味する。理解されるように、借号eとi非は、BSG<sub>T2</sub> 内の対応するカットを形成する。SSGTIの第5及び第 回の実行の開始及び終了を要す。これらの2つのカット c), (i#g11#1, i#2, c), (t4, t4, c), (t3#gt1#1, t3# ュールとピヘイピアのSTGにおける状態境界は対応し ていないため、SSGTI内のカットに対応するBSG72 gt1f0, i#1, 1), (i', i#1, c)である。ここで、エント SGT2内の信号 a 2 が条件condの下で等価であることを は、図12で点線を用いて示されている。これらのカッ 内の「カット」は、等価関係を用いて決定される。例え トは、状備SFを含むスケジュールSTGルーブの第1 1, c)及び(c', c#1, c), (NZ', NZ#1, c), (t4', t4#1, 6のカットに対応してBSGT2内で形成されるカット のメンパである信号に関する等価関係は、(i', i#1, c. c#1, (i", i#2, c. c#1) である。

[0151]上記のことから明らかに示されるように、ルーブ境界変数の多くについての対応は存在しない (例えば、い3'#utl#1、い2'#utl#2、など)。 さらに解析すれば理解されるように、SSG1の大部分 (図中影を付けた部分) はシミュレートされていない。スケジュール内のルーブをもう1回展開することにより、影部分の変数について新たな毎値関係が見出される(すなわち、ルーブを数回の等値関係が現まるためには全部で6回ルーブを原明すると受があることを示すことができる。なお、ピヘイピア内のルーブは、スケジュールを導出する際にスケジューラによって(図然の一致ではなく)ファクタ6でパイプライン化されたことには目すると興味深い。

対応する。この例は2つの理由により興味深い。第1の "Performance analysis and optimization ofschedule s for conditional and loop-intensive specification 配列アクセスは、非解釈関数であるとみなされる。配列 理由は、状態S11及びS12に対応するネストしたル この例は、X. 25通信プロトコルの送信 (send) プロセ インデックスと配列名は対応することが保証される。こ れから得られる正しいスケジュールを図13 (b) に示 す。各状態内の数字は、その状態内で実行される演算に **ープである。第2の理由は、スケジュール内のパスに対** して生成されるデータフローグラフは、ピヘイピアにお いて生成される対応するパスと構造的に同型ではないこ June 1994、をお照。演算への状態の直接的な削当てに (8) に示す。なお、これは配列変数を使用している。 スである。S. Bhattacharya, S. Dey, and F. Brglez, s", in Proc. Design Automation Conf., pp. 491-496, [0152] 4. 7. 2 X. 25通信プロトコル よるピヘイピアに対する制御フローグラフを図13

54 しくないスケジュールを示す。ループのネスティングを 見出すためには、スケジュールのSTGに対する正規変 現(regularexpression)を専門する。Z. Kohavi, "Switc hing and Finite Automata Theory", McGraw Hill Knuk Company, second od., 1978、を事職。

[0153] まず図13 (b) を考える。これに対して 海出される正規変現は50515254 (S354) \* (S55 254 (S354) \*) \* S6である。上付きの\*のある各部 分表現はループ本体を構成する。これは明確にループの ネスティングを確別する。パス列峰がループ本体に遭遇 するたびに、そのループに対する不変項を識別しなけれ ばならない。例えば、節分変現 (S55254 (S 354) \*) \*において、内側ループ (S354) \*の不変項 は、外個ループの不変項を選出するために外側ループに 要求される反復回数と同じ回数だけ舶出されなければな らない。手続きの残りの部分は前と同じであり、第1の ケーススタディの通りに従う。

[0154] 次に、図13(c)を考える。 韓川される 正規表別は、S0S1S254((S3S2S4)\*+(S5S2 S4)\*)\*S6である。この正規表別は、分開ループの内 町に共通の部分表現を有する2個のループを含むという 点で削めものとは異なる。 部分差別 (S3S2A)\*+ (S5S254)\*)\*におはる、外間ループのループ不変 項を放在が変定するまで、外間ループの有限回の反復が評 値されなければならない。 外間ループの有限回の反復が評 値されなければならない。 外間ループの解しの反復が評 値をれなければならない。 外間ループが原設を100で に 内国ループのインスタンス化のすべての可能な組合 せが考慮される。 例えば、外間ループが原分表別 (A\* +B\*)\*である場合、外側ループが解分表別 (A\* +B\*)\*である場合、外側ループが解分表別 (A\*

A\*A\*, A\*B\*, B\*A\*, B\*B\* に沿って変数対応を計算しなければならないことを追求 する。なお、各パスは、それぞれネスティングのない・・ 一ブからなる。外頭ルーブの3回目の実行で、列挙され るパスは次のようになる。

A\*A\*A\*, A\*B\*A\*, B\*A\*A\*, B\*B\*A\*, A\*A\*

B\*、A\*B\*B\*、B\*A\*B\*、B\*B\*B\* 外間ループに対するルーブ不変性を証明するためには、 外間ループの n回目の実行における部分パスPnから生成される変数対応が、Pnから専川される n+1回目の 実行におけるすべての部分パスから生成される対応と同 じままであることを示さなければならない。例えば、上 部の仮設的な例では、外間ルーブの2回の実行において A\*A\*A\*B\*から生成される対応と同じま までなければならない。図13(c)におけるメグジュ ールは、状態53から川る選移の正しくない実現の結 果、実際には正しくない。これは、本発明の事設きでは 被出される。2つの表現におけるねい変数の別に対応を

確立することが不可能であるからである。

とである。図13 (c) に、回じどヘイアアに対する正

(53)

ムは、まず、瞭列された二分木を生成した後、それを正 の下での対応するコード断片に対するスケジュールを図 することができると認識している設計者によって手助で この例は、こ分木敷列(ソート)アルゴリズムのハード ウェア実装である。この2つの部分からなるアルゴリズ しい順序で印字する。このアルゴリズムの一部に対する パく人ガアコード西げや図14 (a) 15 ドナ。 アヘ人 ア アに明確に示されるように、状態S 1における文outp=v al[p]は、状態S2における文outdat=outpの前に実行さ れなければならない。加算器が1個というリソース制約 は、状態S1とS2の文の例の因果関係が保存されるか ぎりこれらの状態を併合することにより1サイクル節約 なされた。なお、状値S1及びS2は2つのループ本体 プで節約されたサイクルの結果、全体で多くのサイクル に現れるため、この最適化には多大な意味がある。ルー 1 4 (b) に示す。この例に対するスケジューリング [0155] 4, 7, 3 二分本胜列

[0156] [4.8 結婚] 本則哲節では、 くんアベ ルビヘイピア仕様のスケジュールの正当性を検証する完 のほとんどのスケジューラに適用可能である。本発明の ュレーションとともに自動帰納法を使用することによっ ジューリング変換が検証可能であり、これにより、既存 アルゴリズムの新規な特徴は、非解釈シンボリックシミ て、データ依存ループを含むループを有する設計を検証 することができることである。 本発明の検証アルゴリズ ムの能力及び実用性は、いくつかの設計について例示し え、校機実行、及びさまざまなルーブ変換のようなスケ **全に形式的な検証技術について説明した。 資算並べ替** らかかわらず、エラーを検出することが可能である。

[0157] 本税明へのこのほかの修正及び変更は本明 **つかの実施倒のみについて特に記載したが、明らかなよ** うに、本発明の技術思想及び技術的範囲から離れること 都許の記載から当業者には明らかである。 本発明のいく なく、さまざまな変更を加えることが可能である。 |図面の簡単な説明|

ジューリングの例であって、エラーが導入されるものを 【図1】クロック境界を挿入することによる単純なスケ

示す図である。

【図2】C責託による回路のどへイビア記述例と、基本 ブロック内における、及び、基本ブロックを超えた、資 群の並べ替えを倒示する対応するスケジュールとを示す |図3| (a) は、ループを含むピヘイピア配述例を示 す図である。(b) 及び(c)は、ループ展開を含む変 煥を示す図である。(d)及び(e)は、それぞれ、ル ープ回転及びループパイプライン化を含む変換を示す図 【図4】 構造グラフの等価性チェックのためのシンボリ ックシミュレーションアルゴリズムの好ましい実施例の

【図5】SCCを抽出し、無別路部分をつぶし、パスを 擬似コードを示す'図である。

列棒するプロセスを示す図である。

|図6】 ピヘイピアコードと、ピヘイピアコードに対応 するパグを有するスケジュールとを示す図である。

[図1] スケジュールをチェックするための、本発明の **算ましい実施倒を示す提似コードの図である。** 

[図8] 対応するスレッドを見つけるために、シンボリ

ックシミュレーションを適用する擬似コードを示す図で

れた。その結果、因果関係はスケジュールに反映されな ルのSTGにおける変数outdatの間の対応を検出しない

は、変数outpの型をrogistorからwireに変更するのを忘 かった。本苑明の手続きは、ピヘイピア及びスケジュー ことにより、この単純なエラーを検出する。なお、この

数を節約することができる。残念ながら、この設計者

【図9】ループからの不変項の抽出を示す類似コードの

図である。

[図10] 例spec#test1に対する投機実行を含むピヘイ ピアSTG及びスケジュールSTGを示す図である。 |図11| 例spec#tost1に対するスケジュールを示す図 である。

である。木苑明のアルゴリズムは、複雑なルーブ構造に

は、ネストした、及び、交遊したループでは非常に複雑

エラー自体は単純であるが、コード断片のルーブ構造

【図12】例spec#test1に対するピヘイピアの構造グラ フを示す図である。 [図13] X25通信プロトコルのsendプロセスを示す 図である。

【図14】二分木盤列の例を示す図である。

【図15】本発明の特徴によるシステムの好ましい実施

別を示す図である。

|図16|| 本発明のもう1つの特徴によるシステムの好 ましい実施例を示す図である。

[符号の説明]

4. 1 ループ不変項抽出器

ツンボリッケツベコフータ 4.

スケジュール及びどへイビア 等価性証明器 15.1 ピヘイピア状態遷移グラフ生成器 ů

スケジュール状態遷移グラフ生成器

'n

スケジュールスレッドセレクタ ピヘイピアスレッドセレクタ 5.4

等価性チェッカ

15.7

30

<u>区</u>

Pl : process

Variable to i integer:

R\_var <- Rinport;

<- Tinport! de\_vaer (= dayport

n\_var (\* Uinport)

wait for Ons

wait until olk-'1' and olk'event; -- CLOCK EDGR

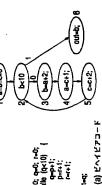
wait until alk-'1' and olk'event; .- CLOCK EDGE to 10 1, war - d. var . dx var . x var; u\_var <- t6 - dr\_ver \* thres \* y\_var P\_var (. P\_var + u\_var \* dx\_var; while (g var < a var) loop R\_VAEE <- R\_VAEE + de\_VAEE

wait for One idoot put toutport (- x\_var) foutport <- T\_var.

loutport (= u\_ver, nd process Ply [函4]

[<u>8</u>3]

[图6]



AMESSON. BSG-) the Ern for IN modus in SBO<sub>T-1</sub>

LOO EQUIYAL ENCETS, e., e 🗍 sale n cach den leyst o'jouin of o ( For cach cany (u,t) is opphibles

OP works way in BSOT sevent layest likes took BDD ander and proper ADD EQUIVALENCE (V. Nag. OWICK

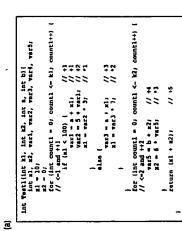
And Poh BSGr

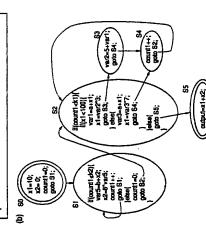
(b) スケジュール STG

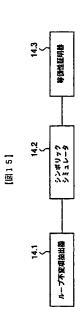
(32)

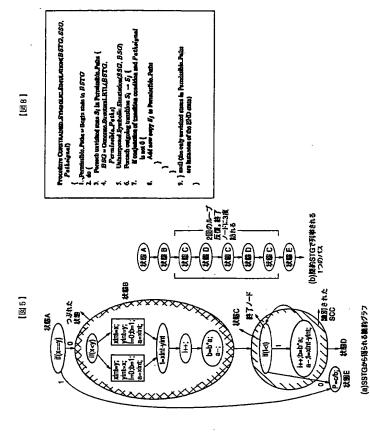
得開2001-142937(P2001-142937A)

[2][2]









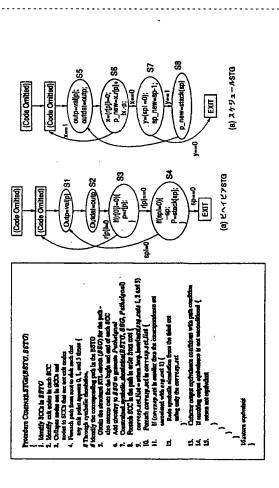
recedure return.loop.invarianti(seg.cuts 1, 2 and 3)

[6図]



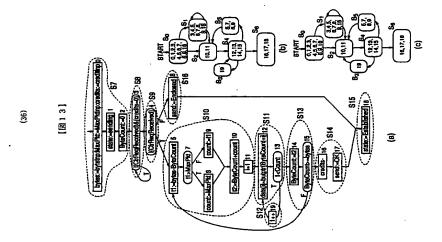
[図14]

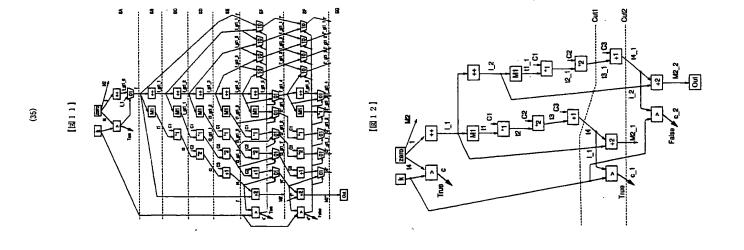
[图7]



[國10]

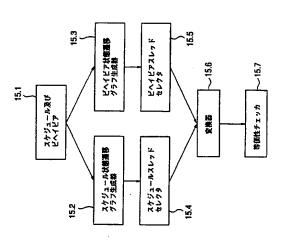
(34)





(37)

[図16]



フロントページの続き

(72)発明者 スプラジット・バタチャリヤアメリカ会の国 ニュージェー

アメリカ合衆間,ニュージャージー 08540 ブリンストン,4 インディベン デンス ウエイ,エヌ・イー・シー・コ

ー・エス・エー・インク内

08540 プリンストン, 4 インディベン

アメリカ合衆国, ニュージャージー

(72)発明者 アナンド・ラグナサン

デンス ウエイ, エヌ・イー・ジー・コ

ー・エス・エー・インク内 (72)発明者 アーティ・グブタ

アメリカ合衆国,ニュージャージー08540 プリンストン,4 インディベン

デンス ウエイ, エヌ・イー・シー・ユー・エス・エー・インク内

ー・エス・エー・インク内 ドターム(参考) 58046 AAO8 BAO3 JAO1 JA04